

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR
GRADO EN INGENIERÍA INFORMÁTICA

DESARROLLO DE UN ENTORNO DE VISUALIZACIÓN DE ALARMAS PARA LA GESTIÓN DE LA RED ELÉCTRICA

Jenifer Prado Pardo
Rosa María Romero Gómez
9 de Julio de 2013

Agradecimientos

A mis padres por aguantar mis nervios con los exámenes, mis cambios de humor y por escuchar cuando les contaba cosas que no entendían del todo. Sin ellos esto no sería posible. A mi hermana, que aunque no lo sepa, ha sido un gran apoyo y un gran motivo para seguir adelante. Al resto de mi familia, porque siempre han estado ahí durante estos años escuchando problemas y alegrías.

A Álvaro por enseñarme a mirar las cosas desde otra perspectiva, por haber sido un apoyo incondicional y por hacer de los últimos años los mejores.

A David del Río por las risas, los lloros y las discusiones durante cuatro años de carrera siendo compañeros de prácticas desde el primer hasta el último día. A Sara por todos los momentos únicos e irrepetibles, por su apoyo y su sabiduría con el Office. A Miguel porque nunca ha perdido la sonrisa y ha hecho que ninguno más la perdiéramos. A David de León y Yolanda, por enseñarme a vivir más despacio y hacer de la nada una buena amistad.

A mi tutora, Rosa, por toda su atención y paciencia durante el desarrollo de este proyecto. A Carlos, por enseñarme a hacer las cosas de una manera buena y bonita. Y, por último, a David Díez, por darme una buena oportunidad, por guiarme y creer en mí.

Resumen

Las redes eléctricas inteligentes, o *Smart Grid*, se caracterizan por su capacidad para integrar y gestionar, en tiempo real, las acciones de los usuarios conectados a la misma, con el fin de conseguir un suministro eléctrico eficiente, seguro y sostenible. Entre las propiedades de este tipo de infraestructura destaca el uso generalizado de sensores y aparatos de telemetría, lo que a los operadores de la red a bregar con grandes volúmenes de datos de muy diverso tipo y naturaleza. Este volumen masivo de información puede llevar al operador a perder, bajo ciertas circunstancias, la consciencia de lo que está ocurriendo en la red de manera global. A fin de minimizar este problema, el presente trabajo fin de grado propone un cambio en el paradigma de diseño de la interfaz de usuario de los sistemas de control de la red eléctrica inteligente.

La solución desarrollada en este proyecto se caracteriza por cuatro aspectos principales. En primer lugar, la división del espacio de trabajo en cuatro zonas: comunicación, monitorización, control y actuaciones. Esta división permite al operador realizar labores de operación de manera simultánea, tratando un evento concreto a la par que mantiene una visión general de la infraestructura. La segunda característica tiene que ver con la coordinación entre las cuatro zonas de trabajo. Gracias a esta característica, el operador pueda tratar un mismo dato desde diferentes puntos de vista. La tercera característica es el uso intensivo de la representación geoespacial de la red. Por último, destaca el uso de capas de información, lo que permite al operador la filtración de los elementos de la red mediante capas de información y el acceso a datos alfanuméricos coordinados con la visualización de la red. La suma de todos estos factores permite al operador realizar las tareas de monitorización y control de diferentes aspectos de la red eléctrica ya que el operador podrá consultar diferentes eventos o datos alfanuméricos sin perder la visión global de la red.

La solución desarrollada se corresponde por un prototipo software de tipo horizontal, desarrollado mediante la tecnología *Windows Presentation Foundation*, y optimizado para su uso en un monitor curvo de 43". Este monitor permite una mayor uniformidad de la pantalla frente al clásico conjunto de pantallas de los entornos de operación. Para la elaboración de esta solución se ha seguido un modelo de proceso iterativo incremental, basado en la realización de sucesivos prototipos, con un nivel de detalle creciente, a fin de alcanzar la solución más adecuada.

La evaluación realizada ha demostrado que la solución cumple todos los requisitos necesarios en el diseño de la interfaz de usuario de un sistema de control. De cara a los trabajos futuros sobre la solución expuesta, se plantea el diseño de un escritorio de crisis. La actual solución plantea las tareas de monitorización y control de la red en situación de normalidad. Sin embargo, en situaciones de crisis, el operador suele enfrentar a avalanchas de alarmas que requieren la adaptación del sistema de control.

Abstract

The Smart Grid is characterized by its ability to be integrated and managed in real time. The purpose of this way of operating is to get a efficient, safe, and sustainable power transmission. One of the most significant features of the Smart Grid is the massive use of sensors and telemetry devices. These components report large volumes of multidimensional information. This large volume of information can compromise, depending on the operational settings, the global situation awareness of the environment. This project proposes a specific user interface design approach to minimize these problems.

The solution developed in this project is characterized by four main aspects. First, the workspace is divided into four areas: communication, monitoring, control, and actions. This structured allows the operator to simultaneously carry out operational tasks: the operator can manage a particular event, maintaining at the same time an overview of the infrastructure. Second, the four work areas are coordinated. Thus, the operator can control the same data from different points of view. The third feature is the extensive use of geospatial representation of grid. Finally, the solution emphasizes the use of information layers, which allows the operator to filter elements of grid. The information layers and access to alphanumeric data are coordinated with the display of grid. Thanks to these features, operators can handle different events or alphanumeric data without losing the overview of grid, allowing them to both monitor and control the grid.

As a solution, a software horizontal prototype based on user interface features has been developed. The solution has been developed by using the Windows Presentation Foundation framework and optimized for being deployed in a 43" curved monitor. This monitor allows more screen uniformity compared to the classic set of screens using in operational environments. Regarding the development process, the solution has been built by following an iterative and incremental model, characterized by the development of successive prototypes. Each prototype provides an increasing level of detail in order to achieve the most appropriate solution.

The evaluation has shown that the solution verifies all requirements. The validated requirements are all required in the design of a user interface of control system. Further work will be oriented to the design of a desktop crisis that support operating task during critical situations. The current solution is designed for monitoring and control the grid in normal situations; however, during crisis situations, the operator has to deal with alarm floods that require the adaptation of the control system.

Índice de contenidos

Resumen.....	3
Abstract	4
Índice de contenidos	5
Índice de figuras	6
Índice de tablas	7
Glosario de términos	8
1 Introducción.....	9
1.1 Problem definition	9
1.2 Aims.....	9
1.3 Stages of project.....	10
1.4 Dissertation structure	10
2 El estado de la cuestión	12
2.1 Revisión de dispositivos de visualización de la información	12
2.2 Revisión de tecnologías para la visualización de la información	15
3 Gestión de proyecto software	23
3.1 Alcance del proyecto	23
3.2 Estimación y presupuesto	23
4 Solución.....	27
4.1 El proceso de desarrollo.....	27
5 Evaluación	63
5.1 Proceso de evaluación.....	63
5.2 Análisis de resultados.....	71
6 Conclusions	72
6.1 Contributions.....	72
6.2 Further work.....	72
6.3 Problems	72
6.4 Personal opinions.....	73
7 Bibliografía	74
Anexo I. Control de versiones	75
Anexo II. Seguimiento de trabajo final de grado.....	76
Anexo III. Especificación de requisitos	77

Índice de figuras

Figura 1: Data Wall de IBM.....	13
Figura 2: <i>Head-Mounted Display</i>	13
Figura 3: Visión proyectada.....	14
Figura 4: Modelo Prototipado.....	28
Figura 5: Prototipo 1.	35
Figura 6: Prototipo Java Rich Client Platform.....	38
Figura 7: Prototipo Windows Presentation Foundation.	39
Figura 8: Disposición final de zonas.	41
Figura 9: Segundo prototipo WPF.....	41
Figura 10: Pila actuaciones.....	42
Figura 11: Visión posicional de la red.....	44
Figura 12: Coordinación vista tabular y visión espacial.	45
Figura 13: <i>Tooltip</i> alarma.	46
Figura 14: Menú radial.	46
Figura 15: Bloc de notas de un elemento.	47
Figura 16: Componente detalle de un elemento.	48
Figura 17: Gráfico de medidas eléctricas.	48
Figura 18: Visualización de simulación.....	49
Figura 19: Sombreado alarma.....	50
Figura 20: Contraste de color en visión tabular.	51
Figura 21: Selección de línea y selección de área.	52
Figura 22: Selección de capas en zona de control.	53
Figura 23: Crear actuación.	54
Figura 24: <i>Log</i> de operaciones.	54
Figura 25: Patrón Model-View-ViewModel.....	56
Figura 26: Organización MVVM.	57
Figura 27: Solución Energos.	58
Figura 28: Estructura de proyectos auxiliares.....	58
Figura 29: Estructura Energos.	59
Figura 30: Carpeta ViewModel.....	59
Figura 31: Carpeta UserControlView.....	60
Figura 32: Carpeta Views.....	60
Figura 33: Carpeta Model.....	61
Figura 34: Carpeta de referencias.	62
Figura 35: Carpeta Utils.....	62

Índice de tablas

Tabla 1: Resumen Java RCP.	18
Tabla 2: Resumen Adobe Flex.	19
Tabla 3: Resumen WPF.....	21
Tabla 4: Comparación global tecnologías.	22
Tabla 5: Estimación de tareas y tiempo.	24
Tabla 6: Coste mensual del personal.	25
Tabla 7: Coste total del personal.....	25
Tabla 8: Coste de materiales.	26
Tabla 9: Costes totales.	26
Tabla 10: Plantilla requisitos.	33
Tabla 11: Relación color y prioridad de alarmas.	50
Tabla 12: Secuencia de acciones con alarma nueva.	51
Tabla 13: Plantilla para escenarios.....	63
Tabla 14: Matriz de trazabilidad entre requisitos y escenarios.	71
Tabla 15: Análisis de resultados de pruebas.	71

Glosario de términos

Alarma: Indicación de incidencia urgente que ocurre en la red eléctrica.

SA: Situation Awareness.

WPF: Windows Presentation Foundation.

RCP: Java Rich Client Platform.

Smart Grid: red inteligente de distribución de energía eléctrica.

Layout: esquema de distribución de los elementos dentro de un diseño.

Mockup: un modelo a escala o a tamaño real del diseño de una interfaz de usuario o de un dispositivo.

1 Introducción

Nowadays, the world works with electrical energy. Many devices use the electrical energy directly, without further conversion; the running of the modern industrial structure depends on the uninterrupted supply of electricity; significant infrastructures, such as rail infrastructure or Internet, are powered by electrical energy. Electrical energy is in a fact a flexibly and healthy form of energy, very easy to carry from one place to another by using conductors. However, this transmission of energy must be optimized to reduce costs and increase productivity, with this purpose the Smart Grid is being developed.

The current electric delivery system (hereafter, “Grid”) is not prepared to coordinate distributed electricity generation and meet future electrical demand needs. In line with this reality, governments and regulators, utility companies, and IT firms are rethinking how the Grid should evolve: the known as “Smart Grid”. The *Smart Grid* is an intelligent electrical grid that uses informatics technology to optimally manage the electrical grid. *Smart Grid* extends the information of the electrical grid to clients and events happened of the grid.

The operation of the *Smart Grid* is performed within specific control rooms. In these control rooms, a group of operators are in charge of monitoring and controlling the status of the electrical grid by using a specific operational system. This project is related to this specific operational system.

1.1 Problem definition

The problem faced in this work is related to the operation of the Smart Grid within control rooms. The continuous sensing of the grid state, the increasing interconnectivity and complexity of the infrastructure, and the rising amount of operational information to manage require innovative control systems that support new operational models, based not only on identifying breakdowns but also on anticipating and diagnosing them. The Smart Grid requires the operators manage a great volume of information at quasi-real time, compelling operators to perceive and comprehend new events in order to properly manage them.

This project is oriented to provide informatics technologies that allow operators to control the Smart Grid in a comprehensive way. This problem encompasses several challenges: operators should perform a set of parallel tasks to control the infrastructure; the information provides by the system is characterized as multi-source and multidimensional; and, finally, operators do not work in an isolated way but interacting with others operators.

1.2 Aims

The purpose of this project is to assist the labor of the Smart Grid operators. With such a purpose, the main aim of the work is to re-design the user interface of an operational system for the *Smart Grid*. This aim may be divided into the following goals:

- The solution must allow the operator to simultaneously control and monitor the electrical grid. The operator must know what happened at every moment in order to maintain the “normal state” of grid.

- The current technology provides alphanumeric representations of the information, integrating different types of representations. The information is multidimensional type. The solution must provide different views to ease.
- The results from state measurement shall be combined with a wide-area geographical information system. The wide-area geographical information system shall cover a broad region including the control center's own service territory as well as all interconnected areas. This will increase the situational awareness across a very broad scope and prevent inappropriate operations when neighboring system is not fully known. Then the solution must provide geographical representations of information.

1.3 Stages of project

The development of this project has been divided into four phases.

- **Phase 1. Studying the context.** The first state of the project was to scope and review the context. The study is divided into two aspects of the solution. The first studied is a possible hardware components. The second studied is the possible technologies to use.
- **Phase 1. Definition of the problem.** The ideas and features that the researcher provided are the basis of the analysis. The result of this analysis is the set of requirements of the solution to the studied problem.
- **Phase 2. Elaboration of the solution.** In this phase, the principal aim is to design the solution to the problem. The solution must meet the requirements defined in the second phase.
- **Phase 3. Evaluation of the solution.** In this phase, the aim is to validate the solution previously designed. The validation encompasses both the endorsement of the system requirements and the usability evaluation of the solution.

1.4 Dissertation structure

This document exposed the memory of the final dissertation of the Computer Science degree. The structure of the memory is divided into four sections related to the four stages of project.

The first section presents the study of the context. This study covers the review of different rich-user interface technologies as well as the revision of display technologies in which the system could be deployed. Regarding the rich-user interface technologies, the review has considered the following: Java Rich Client Platform, Adobe Flex, and Windows Presentation Foundation.

The third section presents the requirement's capture and a viability study of the management of project. The requirement's capture includes some ideas provided by the researched. Besides, in this section the scope of the project is defined. Section four explains the design technique chosen from the various ones available. The design of solution consists of a prototype's model. The development of the solution has been in four iterations. The solution is centered on the design of user interface. Each prototype of the solution is the result the one iteration. In the fourth iteration

obtain the final solution. The design of solution consists of four iterations. Each prototype of the solution is the result the one iteration. In the fourth iteration obtain the final solution.

Section five describes the evaluation of solution. Such an evaluation is based on tasks path. Finally, personal opinions and conclusions are compiled in the last section of the dissertation.

2 El estado de la cuestión

Endsley define de forma general el *Situation Awareness* (SA): “The perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future”. Teniendo en cuenta esta definición el SA es un factor muy importante para el contexto del proyecto, un operador en una sala de control de la red eléctrica. Sin embargo, debido al crecimiento de la complejidad y la interconectividad de la red eléctrica, el hecho de que el operador adquiriera el SA se ha vuelto más complicado. En particular se centra este crecimiento de complejidad en el volumen y tipología de datos a manejar por el operador.

Luego, la revisión de dispositivos, que a continuación se expone, se centrará en los dispositivos de visualización de información. Esta revisión tiene por fin determinar el dispositivo más adecuado para soportar el alcance del SA en el contexto de operación de una sala de control.

Por otro lado, se realizará una revisión de tecnologías para el desarrollo de visualizaciones de alarmas que permitan percibir y comprender el estado de la infraestructura de la red eléctrica.

2.1 Revisión de dispositivos de visualización de la información

Los dispositivos de visualización son todos aquellos dispositivos físicos que se pueden conectar a un ordenador y que muestren visualmente al usuario la salida que emite el ordenador.

En la sección anterior se ha indicado la importancia de analizar los diferentes dispositivos que pueden servir de herramienta principal para el operador de una sala de control.

El estudio de los dispositivos de visualización parte de tres tipos: dispositivos dinámicos, dispositivos de inmersión y dispositivos estáticos.

Dispositivos de visualización dinámicos

Los dispositivos de visualización dinámicos son todos aquellos dispositivos que permiten que el usuario interaccione con ellos sin mediación de ningún otro dispositivo extra, tal como una pantalla *multi-touch* o un *DataWall*, como el que se muestra en la [Figura 1](#). Estos dispositivos están enfocados a una mayor cooperación y un alto nivel de interacción. Se plantea así un entorno colaborativo, en el que varios usuarios podrían intercambiar, no sólo opiniones sino también estrategias a seguir a la hora de realizar una actividad.



Figura 1: Data Wall de IBM¹.

Dispositivos de visualización de inmersión

Los dispositivos de inmersión son aquellos que permiten que el usuario interactúe con una realidad aumentada. Estos son dispositivos como los *Head-Mounted Display* (HMD), que se muestra en la Figura 2, o *Spatial Displays*, proporcionan un mayor detalle de la zona concreta dónde el usuario realiza el control. Esto es una ventaja de cara a poder simular diferentes tipos de eventos y generar planes de acción ante esos eventos, pudiendo comprobar así los resultados en “primera persona”.



Figura 2: Head-Mounted Display².

¹ <http://vimeo.com/51454262>

² <http://commons.wikimedia.org/wiki/File:Aimoneyetap.jpg>

Dispositivos de visualización estáticos

Los dispositivos estáticos son aquellos dispositivos que ofrecen una visualización plana y una interacción con el usuario mediante dispositivos externos, un ejemplo de estos dispositivos son las pantallas de ordenadores comunes.

Dentro de los dispositivos de visualización estáticos se encuentran dos nuevas categorías a tener en cuenta: los sistemas de visión directa y los sistemas de visión proyectada.

Los sistemas de visión proyectada, como los proyectores tradicionales, como el que se muestra en la [Figura 3](#), o los proyectores láser en 3D, son sistemas que permiten proyectar la vista de la zona específica pero puede incomodar la interacción con los eventos surgidos o los datos de la misma.

Los sistemas de visión directa, como los monitores tradicionales o paneles planos, ofrecen la representación de la imagen completa, al igual que los visionados con proyección, y ofrecen una mayor comodidad para la interacción entre los datos, la visión global y el operario.



Figura 3: Visión proyectada³.

³ <http://decoestilo.mujerhoy.com/articulo/claves-para-comprar-un-proyector/>

Elección del dispositivo de visualización

Los dispositivos estáticos, en contra de los dispositivos dinámicos, ofrecen una menor capacidad de interacción, y, en contra de los dispositivos de inmersión, no proporcionan un mayor realismo que el máximo que puede ofrecer una pantalla. Sin embargo, los dispositivos estáticos son los más adecuados para tener un conocimiento global de la situación de la red. Estos dispositivos tienen mayor capacidad para: tener vigilancia de la zona específica y tener una mayor interpretación de los datos y los hechos acaecidos. Los dispositivos dinámicos y de inmersión ofrecen una mayor interacción por lo que están orientados a entornos colaborativos pero en el caso de este proyecto el contexto se basa en el entorno individual de un operador, esta individualización es potenciada en los dispositivos estáticos.

Por otro lado, el contexto del proyecto exige el manejo de un gran volumen de datos esto se vuelve ventaja en el caso de los dispositivos estáticos, puesto que las tareas a que se puedan realizar sobre una serie de datos resultan más ágiles en este tipo de dispositivos si se compara con los dinámicos y de inmersión.

En conclusión, los dispositivos estáticos ofrecen una mejor comprensión desde diferentes aspectos (imágenes, información textual o alfanumérica, etc.) de los eventos que puedan darse. Por ello el estudio en profundidad se realiza en este tipo de dispositivos.

Dentro de los dispositivos estáticos y, en el ámbito de un operador trabajando en una sala de control, se suele encontrar con que utilizan del orden de entre 3 y 6 monitores conectados al ordenador. De esta forma, los operadores pueden tener abiertos diferentes programas que necesiten: el correo interno de la empresa, la visión global de la red, el escritorio para poder ejecutar acciones sobre eventos o zonas de la red, etc. Esta disposición de pantallas tiene muchas ventajas en cuanto a la apertura de diferentes programas para trabajar. Sin embargo, si se compara la posibilidad de trabajar con 3 monitores conectados o un monitor curvo de 43" se encuentran desventajas para la primera opción. En un estudio interno de la Universidad Carlos III de Madrid [1] acerca de las características y la funcionalidad que ofrece el monitor Ostendo CRVD 43" a los usuarios, se concluye resaltando la agilidad que tiene el usuario a la hora de realizar ciertas acciones: búsqueda de datos.

2.2 Revisión de tecnologías para la visualización de la información

El objetivo de este proyecto es desarrollar un entorno de visualización de alarmas que permita alcanzar un adecuado SA para el operador. Con este objetivo, se realiza una revisión de tecnologías que provean altas capacidades de interacción y visualización. La importancia de la revisión de las tecnologías reside en que la tecnología será la que establezca los límites para los componentes visuales. El análisis permitirá conocer la tecnología más apropiada para los requisitos que presenta la interfaz.

Al tratarse de una labor en tiempo real, la que realiza el operador, enmarcada en un contexto crítico, se deben revisar tecnologías que permitan el desarrollo de un entorno de visualización de escritorio. Por lo tanto, las tecnologías a analizar son: Windows Presentation Foundation (WPF), Java Rich Client Platform y Adobe Flex junto con Adobe Air.

Criterios de evaluación

Para cada una de las tecnologías propuestas se van a estudiar una serie de características y según los resultados obtenidos, saldrá elegida la tecnología que mejor se adapte a este proyecto.

La primera característica que se deberá analizar es el **coste** de las licencias y herramientas necesarias para el desarrollo. Esta característica se medirá con valores:

- **Gratuito:** en caso de que no tenga coste.
- **Privado:** en caso de que si tenga coste.

De cara al desarrollo, se tendrá en cuenta la **curva de aprendizaje** que tiene cada una de las tecnologías seleccionadas. Considerando como curva de aprendizaje el tiempo necesario y el esfuerzo que ha de realizar cada desarrollador para poder tener unos conocimientos altos y un manejo fluido de la tecnología. Este criterio se medirá con los siguientes valores:

- **Alta:** si el esfuerzo requerido es muy alto, con un tiempo estimado de aprendizaje del orden de 3 meses o más de aprendizaje.
- **Media:** si el esfuerzo requerido es lo normal para un desarrollador que no conoce la tecnología, con un tiempo de aprendizaje aproximado a 1 mes.
- **Baja:** si el esfuerzo requerido es escaso ya que el desarrollador o bien conoce la tecnología o bien tiene conocimientos sobre lenguajes parecidos, con un tiempo de aprendizaje aproximado de 2 semanas.

Partiendo de la curva de aprendizaje que puede tener cada una de las tecnologías, se apreciará tanto la existencia de **material de ayuda y referencias** para su consulta como el tamaño de la comunidad de desarrolladores. En este caso se valorará el tipo de material que se encuentra de la siguiente forma:

- **Libros:** en caso de que el material que se ofrezca sea en formato de libro.
- **Webs:** en caso de que el material que se ofrezca sea en formato de webs de ayuda para desarrolladores.
- **Tutoriales:** en caso de que el material que se ofrezca sea en formato de guías para la realización de ejemplos.

Igualmente, se considera un dato de importancia el **número de bibliotecas públicas**, principalmente, aquellas útiles para la creación y modificación de componentes visuales para la interfaz. Este criterio se valorará de la siguiente forma:

- **Alto:** si el número de bibliotecas públicas que se encuentra es aproximadamente superior a 100.
- **Medio:** si el número de bibliotecas públicas que se encuentra es aproximadamente entre 50 y 100.
- **Bajo:** si el número de bibliotecas públicas que se encuentra es aproximadamente menor que 50.

Debido a que se trata del desarrollo de una aplicación que trabaja sobre un mapa, otra gran característica a tratar será la **integración de mapas** y la información que se proporcione para la interacción con ellos en las diferentes plataformas. Este criterio se medirá con los siguientes valores:

- **Si:** si permite la integración de diferentes librerías externas de mapas.
- **No:** si no permite la integración de ningún tipo de librería externas de mapas.

Puesto que se trata del desarrollo de una interfaz de usuario, un criterio muy importante a tener en cuenta, es la capacidad de crear y modificar los **componentes visuales**. Para esta solución es importante tener una gran variedad de componentes visuales básicos de cada tecnología y, también, el poder generar los componentes visuales que se puedan necesitar. Este criterio se medirá con los siguientes valores:

- **Alto:** si la tecnología permite la creación de componentes que no hereden de componentes básicos.
- **Medio:** si la tecnología permite la creación y modificación de componentes que hereden de componentes básicos.
- **Bajo:** si la tecnología sólo ofrece la posibilidad de utilizar los componentes básicos.

Puesto que se van a tratar medidas numéricas, es importante que la tecnología ofrezca **librerías de gráficos**. Estas librerías serán necesarias para la implementación de componentes de gráficos de datos. Este criterio se medirá con los siguientes valores:

- **Si:** si la tecnología ofrece librerías de gráficos.
- **No:** si la tecnología no ofrece librerías de gráficos.

El último criterio a tener en cuenta referencia a la capacidad de interacción que ofrezca la tecnología, **interacción rica en la interfaz**. Este criterio evalúa las posibilidades de interacción que tiene una tecnología, por ejemplo si se puede utilizar: *DragAndDrop*, ventanas movibles dentro de la misma interfaz, etc. Los valores para medir este criterio serán los siguientes:

- **Alta:** si la tecnología ofrece crear y modificar diferentes posibilidades de interacción.
- **Media:** si la tecnología ofrece variaciones de las posibilidades básicas de interacción.
- **Baja:** si la tecnología ofrece sólo las posibilidades básicas de interacción.

Java Rich Client Platform

Rich Client Platform se basa en el desarrollo de aplicaciones que sirvan como arquitectura y plataforma para la integración de otras tecnologías en la aplicación mediante los conocidos plug-in. De esta misma forma, esta tecnología sirve para crear diferentes plug-in que se integren en aplicaciones Java.

El entorno de desarrollo es gratuito, sólo se necesita el software *Eclipse* que se obtiene mediante descarga gratuita de la página web oficial. Otro requisito imprescindible es tener mínimo instalada la versión de Java 1.6, suministrada por la empresa Oracle, también de forma gratuita.

El desarrollo de las aplicaciones Rich Client Plataform, se basa en la conjunción de diferentes frameworks de Java. Esto es una desventaja clara a la hora de analizar la curva de aprendizaje, ya que se deberá estudiar gran parte de los servicios que ofrecen los frameworks y cómo se utilizan. Otro punto fuerte en la curva de aprendizaje resulta ser la estructura rígida que presentan las aplicaciones RCP, luego también se deberá aprender a diseñar según esta estructura. Por el contrario, y como punto positivo, el lenguaje de la tecnología se basa en Java y en la orientación a objetos. Luego la curva de aprendizaje que tiene esta tecnología se presenta con varios picos de dificultad a medida que se desarrolla la aplicación y se tocan diferentes partes de la misma. A esta dificultad no se le puede sumar como ayuda las referencias o ejemplos que existen puesto que la documentación de esta tecnología resulta casi nula tanto en el formato web como en el formato papel.

En el caso de las bibliotecas que se puedan utilizar en el desarrollo de la aplicación no se encuentra una situación diferente, ya que la tecnología resulta tan cerrada de cara a la evolución de Java que sólo se pueden utilizar las bibliotecas propias los frameworks con los que trabaja la tecnología por defecto.

La integración de mapas resulta ser también cerrado a la cantidad de variedad de mapas que se pueden encontrar en la actualidad. Las aplicaciones RCP sólo pueden integrar mapas que, como se ha comentado en el párrafo anterior, vienen integrados en los frameworks por defecto. En este caso se tendría que utilizar *Standard Widget Toolkit (SWT)*, conjunto de herramientas para las interfaces en Java, y la librería *Swing*. La tecnología Java RCP resulta muy arcaica en cuanto a componentes visuales se refiere. Únicamente permite utilizar los componentes básicos de Java, y la modificación de ellos resulta en ocasiones muy costosa y en otras ocasiones no se puede aplicar. Ocurre la misma restricción que con los componentes visuales en el plano de las librerías de gráficos y la interacción posible en la interfaz. En ambos criterios la tecnología se acota a las posibilidades más básicas y propias de la misma tecnología. Los gráficos a mostrar son propios de una librería de Java y no permiten mayor modificación que los datos a mostrar y el tipo de gráfico. En cuanto a la interacción, el 90 % de la interacción posible se basa en un simple click sobre un botón u objeto. En conclusión, se puede resumir cada una de las características de la tecnología en la siguiente tabla:

Características:	Valor:
Coste y licencias	Gratuito
Curva de aprendizaje	Baja
Material de ayuda y referencias	Webs
Número de bibliotecas públicas	Bajo
Integración con mapas	No
Componentes visuales	Bajo
Librerías de gráficos	Si
Interacción rica en la interfaz	Baja

Tabla 1: Resumen Java RCP.

Adobe Flex junto con Adobe Air

Adobe Flex es una tecnología basada en la plataforma Flash, orientada a los servicios web. Para poder desarrollar una aplicación de escritorio es necesario que se ejecute sobre el software Adobe Air. Esto hace que sea necesaria la licencia de Adobe Flash Builder, la cual no es gratuita. Como se puede observar en la imagen, con este software ya se permitiría la opción de creación de un proyecto para una aplicación de escritorio.

El lenguaje en el que se basa Adobe Flex está dividido en dos partes: la parte de funcionalidad está basada en el lenguaje Java, y la parte de interfaz está basada en los lenguajes XML y HTML. Los tres lenguajes son muy corrientes en el desarrollo de aplicaciones por lo que la curva de aprendizaje no requiere mayor esfuerzo que el aprender las peculiaridades que pueda ofrecer Flex.

En cuanto a la documentación, no existe una gran variedad ya que la tecnología está quedando obsoleta debido a la aparición de HTML5. Por el contrario, existen muchas librerías de carácter público que incluyen una gran variedad de componentes gráficos.

En el caso de la integración de mapas, Flex ofrece una mayor libertad para la inclusión de diferentes servidores de mapas.

Esta tecnología tiene una gran variedad de componentes visuales básicos y, además, permite la modificación o variación de ellos.

Adobe Flex es una tecnología en la que se pueden integrar multitud de componentes, entre ellos existen diferentes librerías que permiten la visualización de gráficos de estadísticas.

El último criterio a estudiar, interacción rica en la interfaz, se cumple de manera moderada puesto que esta tecnología permite mezclar y modificar los modelos de interacción básicos.

En resumen, se puede apreciar los puntos fuertes y débiles de esta tecnología en la siguiente tabla:

Características:	Valor:
Coste y licencias	Privada
Curva de aprendizaje	Media
Material de ayuda y referencias	Webs, Tutoriales
Número de bibliotecas públicas	Bajo
Integración con mapas	Si
Componentes visuales	Medio
Librerías de gráficos	Si
Interacción rica en la interfaz	Media

Tabla 2: Resumen Adobe Flex.

Windows Presentation Foundation

Esta tecnología pertenece a la empresa Microsoft, por lo que para poder utilizarla es necesario tener instalado el último framework de Windows disponible. También se necesita tener instalado la estructura para aplicaciones web Microsoft Silverlight. Estos dos requisitos son gratuitos y sólo se pueden obtener si el sistema operativo sobre el que se desarrolla es Windows.

Para poder desarrollar aplicaciones en WPF es necesario un software específico que prepare el entorno de desarrollo, este es el programa Visual Studio. Este software también pertenece a Microsoft y la licencia del mismo no es gratuita.

WPF separa la parte visual y gráfica de la parte funcional. En el caso de la parte visual se ha de utilizar el lenguaje XAML, este lenguaje sólo requiere esfuerzo para conocer los componentes específicos de esta tecnología, y si desarrollador tiene conocimientos previos de XML o HTML disminuye el esfuerzo para aprender el nuevo lenguaje. En el caso de parte funcional se ha de utilizar C#, en este caso el desarrollador necesitaría más tiempo para aprender y comprender este lenguaje de programación.

A pesar del aprendizaje que conllevan los diferentes lenguajes de programación, el mayor esfuerzo recae sobre la comprensión del patrón de diseño que tiene WPF. Esta tecnología impone un nuevo patrón de diseño llamado Model-View-ViewModel. Este patrón resulta muy diferente a lo corriente en este tipo de aplicaciones: Modelo - Vista – Controlador, por lo tanto, es en el aprendizaje de la estructura de la aplicación dónde mayor esfuerzo se requiere.

Al tratarse de una tecnología soportada por Microsoft y teniendo en cuenta el número de dispositivos que acapara esta empresa, el desarrollo en WPF es muy alto por lo que en la web se encuentran numerosos ejemplos que sirven de ayuda. También existe la web oficial para desarrolladores de Microsoft [2] y la web [3] en la que sólo se explican problemas relacionados con WPF.

Al igual que existe una gran comunidad de desarrolladores existen también numerosas bibliotecas públicas y privadas.

En cuanto a la integración de mapas con la tecnología no existe mayor problema que las licencias que puedan suponer los mapas. WPF no sólo tiene una librería para mapas propios de Microsoft sino que también pueden incluirse diversos mapas de otras empresas como: GoogleMaps, ESRI, etc.

Esta tecnología presenta una gran libertad a la hora de la creación de componentes visuales. No sólo permite la utilización y variación de componentes básicos, sino que también permite generar nuevos componentes según las necesidades del diseñador.

Por otro lado, existen varias librerías, externas, para la implementación de gráficos de estadísticas.

La interacción que ofrece esta tecnología tiene un gran abanico de posibilidades como interacción básica y también permite la modificación o creación de posibilidades de interacción, por lo que se trata de una tecnología muy rica en interacción en la interfaz de usuario.

A continuación, se muestra en la [Tabla 3](#) el resumen de las características de WPF:

Características:	Valor:
Coste y licencias	Privado
Curva de aprendizaje	Alta
Material de ayuda y referencias	Libros, webs, tutoriales
Número de bibliotecas públicas	Alto
Integración con mapas	Si
Componentes visuales	Alto
Librerías de gráficos	Si
Interacción rica en la interfaz	Alta

Tabla 3: Resumen WPF.

Comparación específica

Tras haber descrito cada una de las tecnologías por separado, se llega a la conclusión de que las dos tecnologías más óptimas para el desarrollo de este proyecto pueden ser: Windows Presentation Foundation o Adobe Flex.

Adobe Flex permite la creación y modificación de componentes visuales muy similares a los que ofrece Flash, esto también lo ofrece WPF con la instalación de Microsoft Silverlight. La tecnología WPF está orientada a la creación de interfaces con componentes interactivos completamente modificables, mientras que Adobe Flex obtiene casi todos los componentes de las librerías de Flash. WPF también permite la creación de componentes propios del desarrollador. Además, para facilitar esta tarea en el entorno de desarrollo se incluye un software de diseño que únicamente tiene por objeto los cambios de estilos de los componentes.

Por otro lado, y a diferencia de Adobe Flex, WPF tiene una separación entre contenido y estilos en cada uno de los componentes que se utilicen en la aplicación. Esta separación agiliza la funcionalidad propia del componente, y permite obtener componentes derivados de uno inicial sin tener que realizar cambios en el modelo de datos.

Otra desventaja de Adobe Flex es que ha perdido mercado en los últimos años, por lo que la tecnología está quedando obsoleta para el desarrollo de aplicaciones.

A pesar de que WPF tiene una curva de aprendizaje más alta, al comparar estas tecnologías de una forma más específica y más orientada a las necesidades del proyecto como son las capacidades de componentes visuales y de interacción.

Luego, en conclusión y tras la comparación de estas tres tecnologías se obtiene una única tecnología óptima para el desarrollo de este proyecto: *Windows Presentation Foundation*. La elección está basada en las características visuales y de interacción, además de las características de estructura, de la tecnología WPF.

A continuación se muestra una tabla de comparación global de las tres tecnologías.

Características:	Java RCP:	Adobe Flex:	WPF:
Coste y licencias	Gratuito	Privada	Privado
Curva de aprendizaje	Baja	Media	Alta
Material de ayuda y referencias	Webs	Webs, Tutoriales	Libros, webs, tutoriales
Número de bibliotecas públicas	Bajo	Bajo	Alto
Integración con mapas	No	Si	Si
Componentes visuales	Bajo	Medio	Alto
Librerías de gráficos	Si	Si	Si
Interacción rica en la interfaz	Baja	Media	Alta

Tabla 4: Comparación global tecnologías.

3 Gestión de proyecto software

En esta sección se va a desarrollar el presupuesto contemplado para la realización de este proyecto. En primer lugar se definirá el alcance que conlleva el proyecto. Partiendo de este alcance se detallará una estimación ajustada a las necesidades del proyecto, la estimación estará basada en las tareas a realizar y los recursos necesarios para poder completar estas mismas.

Por último, se desglosará el presupuesto que se obtiene según la estimación obtenida. En este desglose se incluirán detallados los gastos de personal, de recursos y generales.

3.1 Alcance del proyecto

Definición del proyecto

El proyecto a desarrollar es un proyecto que forma parte del proyecto de investigación **Energos** [4]. El proyecto de investigación **Energos** tiene por objetivo el desarrollo de conocimientos y tecnologías que permitan avanzar en la implantación de redes inteligentes de distribución de energía eléctrica (*Smart-Grid*). El proyecto que aquí se propone sólo acoge una parte del proyecto **Energos**, concretamente la visualización de datos y el diseño de la interfaz de usuario. Luego, el objetivo de este proyecto es diseñar y obtener una interfaz de usuario apta para el contexto establecido: un operador en una sala de control de red eléctrica. Además, este proyecto se incluye en un contexto de desarrollo universitario.

3.2 Estimación y presupuesto

Estimación de tareas y recursos

En el desarrollo de un prototipo software que se centra en la interfaz de usuario se tienen las siguientes tareas:

- **Definición de requisitos:** en esta tarea será necesario un análisis del contexto en el que se quiere implantar la interfaz de usuario. En términos de tiempo puede requerir, aproximadamente, dos meses.
- **Definición de tecnología:** esta tarea tiene por objetivo el aprendizaje de las diferentes tecnologías y la comparación de resultados entre ellas. En términos de tiempo, y contando con dos personas al cargo de esta tarea, puede requerir aproximadamente dos meses.
- **Implementación:** esta es la tarea de mayor coste en cuanto a tiempo puesto que es el periodo en el que se ha de implementar todo el diseño extraído del análisis. Aproximadamente, contando con dos personas para la realización de esta tarea, se necesitarán cuatro meses.

Para llevar a cabo estas tareas serán necesarios los siguientes perfiles:

- **Investigador principal:** persona con título de doctor, que realice la mayor parte del análisis y estudio del contexto de la solución. Una vez llevada a cabo esta tarea, este recurso sólo se dedicará a la gestión.

- **Ingeniero superior:** persona titulada, que realice todo el diseño de la solución y parte de la implementación.
- **Becario:** persona aun sin titular, que se dedique exclusivamente a la implementación.

Al establecer una relación entre tareas y recursos humanos se puede estimar el tiempo que será necesario para obtener la solución final. En la siguiente tabla se muestra cómo deberán asignarse las tareas.

Tareas asignadas	Recursos humanos	Tiempo estimado
Análisis	Investigador	2 meses
Gestión del proyecto	Investigador	Vida del proyecto
Diseño	Ingeniero superior	2 meses
Implementación	Ingeniero superior Becario	4 meses

Tabla 5: Estimación de tareas y tiempo.

En conclusión, y siguiendo la asignación de tareas definida, se estima que la duración del proyecto sean 8 meses.

Presupuesto

En esta sección se desglosa cada uno de los apartados necesarios para la vida de un proyecto: personal, material y gastos generales. Todos los costes aquí propuestos se basan en una duración de proyecto de 8 meses.

En el apartado de personal y, teniendo en cuenta que se trata de un proyecto que forma parte de un proyecto de investigación, se incluye a un investigador principal que será el supervisor de todo el proyecto. Al tratarse de un proyecto vinculado con la Universidad Carlos III de Madrid, el investigador principal tendrá el título de doctor y pertenecerá a la docencia de ésta universidad. El investigador principal abarcará el rol, dentro del proyecto, de jefe de proyecto.

Además del investigador principal, se necesitará a un ingeniero superior contratado y a un becario. El ingeniero superior abarcará dos roles, dentro del proyecto, diseñador y programador. En el último caso, el becario, sólo tendrá el rol de programador.

Debido a que el objetivo del proyecto es el diseño de una interfaz óptima para un entorno concreto, el ingeniero superior tendrá una carga mayor de horas en el proyecto.

En la siguiente tabla se muestra el coste mensual, habiendo aplicado la tasa de cotización de la Seguridad Social con valor de 23,6%.

Personal	Coste Hora	Horas/Semana	Salario Bruto/Mes	Coste Seguridad Social/Mes
Investigador Principal	15 €	8	480 €	113,30 €
Ingeniero superior	13 €	25	1.287 €	303,71 €
Becario	5 €	20	380 €	89,68 €

Tabla 6: Coste mensual del personal.

En la [Tabla 7](#) se muestra el coste total del personal para el proyecto con una duración de 8 meses.

Personal	Coste total (8 meses)
Investigador principal	4.747,01 €
Ingeniero superior	12.724,94 €
Becario	3.757,44 €

Tabla 7: Coste total del personal.

En el apartado de material, se ha de tener en cuenta que el trabajo y diseño de la interfaz se realiza sobre la pantalla Ostendo CVRD 42". Por lo tanto, es imprescindible tener este monitor para poder probar los diseños. La conexión que esta pantalla requiere es HDMI, luego se necesita un cable Philips HDMI-HDMI para que la imagen de la interfaz tenga la mayor calidad posible. Debido al carácter de prueba que tiene este monitor, se necesita un ordenador adaptado a este monitor y que sólo sea utilizado para ejecutar pruebas. Las necesidades del monitor influyen también en el ordenador, se necesitará una tarjeta gráfica AMD Radeon HD 6570 para poder visualizar cualquier prueba en el monitor curvo.

En total son necesarios cuatro ordenadores, uno de pruebas y tres para el uso completo del personal. Los cuatro ordenadores serán el mismo modelo: HP Pavilion p6-2322es. Debido a que el dispositivo para el que se implementa (monitor curvo 42") tiene unas dimensiones que distan mucho de los monitores de ordenador comunes, se necesitan monitores que ofrezcan menor diferencia a la hora de ejecutar las pruebas con respecto al monitor curvo. Se necesitan tres monitores modelo HP Pavilion 22xi.

El entorno de desarrollo para la tecnología Windows Presentation Foundation es el software Visual Studio 2012 de Microsoft. Se necesitan cuatro licencias de este software en la versión Professional. Además, para la documentación del proyecto y los posibles tratamientos de datos serán necesarias las licencias de Microsoft Office 2010 Home and Business. Por consecuencia de la documentación y los posibles trámites que se puedan realizar será necesaria una impresora HP Officejet 4620.

Sobre el coste de estos materiales se aplicará una amortización con un valor de 48, exceptuando el coste de las licencias.

A continuación, en la [Tabla 8](#) se detalla, de cada uno de los materiales especificados, el coste, el número de unidades, el coste total bruto y el coste total sobre el proyecto aplicando la amortización o coste imputable.

Material	Coste	Unidades	Coste total	Coste imputable
Ordenador HP Pavilion p6-2322es	599,99 €	4	2.399,96 €	399,99 €
Monitor HP Pavilion 22xi	158,99 €	3	476,97 €	79,50 €
Monitor Ostendo CVRD 42"	4.991,19 €	1	4.991,19 €	831,87 €
Impresora HP Officejet 4620	86,82 €	1	86,82 €	14,47 €
Tarjeta gráfica AMD Radeon HD 6570	51 €	1	51,13 €	8,52 €
Cable HDMI Philips	34 €	1	34 €	5,67 €
Licencia Visual Studio 2012 Professional	1.477 €	4	5.908 €	5.908 €
Licencia Microsoft Office 2010 Home and Bussines	76,80 €	4	307,20 €	307,20 €

Tabla 8: Coste de materiales.

El último aspecto a tener en cuenta en el presupuesto del proyecto son los gastos generales como luz, agua, etc. Al tratarse de un proyecto de investigación dentro de la Universidad Carlos III de Madrid el coste de estos gastos suponen el 5% del coste total del proyecto. Luego, el 5% se aplica sobre la suma del coste total del personal y del coste total del material.

En la [Tabla 9](#), se muestra los costes totales de los tres aspectos especificados y el coste total del proyecto.

Aspectos económicos	Tarifa mensual
Coste personal	21.229,39 €
Coste material	7.555,21 €
Gastos generales	1.439,23 €
Total proyecto	30.223,83 €

Tabla 9: Costes totales.

4 Solución

El objetivo de este proyecto es asistir a la adquisición de SA a través de un entorno de visualización de alarmas. Por lo que la solución que se plantea es el resultado del diseño de una interfaz que permita al operador ser capaz de tratar un gran volumen de datos sin perder consciencia de lo que ocurre en la red eléctrica. La interfaz de usuario resultante se ha de dividir en 4 zonas: Comunicación, Monitorización, Control y Actuaciones. La división de estas 4 zonas permitirá al operador poder trabajar sobre todo el entorno sin tener que prescindir de alguna de sus tareas. La zona de comunicación englobará un sistema de mensajería asíncrono. La zona de actuaciones contendrá las diferentes tareas que debe realizar el operador. La zona de control será el contenedor de los detalles que el operador desee ampliar, incluidas las actuaciones que desee revisar. Por último, la zona de monitorización ofrecerá al operador una visión global de la red de forma espacial y alfanumérica. Esta zona permitirá consultar: la visión espacial de la red, la visión posicional, datos relativos a cada elemento de red, medidas de valores eléctricos, etc. Esta zona también será la encargada de resaltar y llamar la atención del operador cuando una alarma nueva se produzca.

En esta sección se detalla todo el proceso seguido para obtener la solución. La estructura que sigue este apartado es la estructura lógica para explicar el proceso de desarrollo. En primer lugar se plantea el tipo de modelo de vida de software que se sigue a lo largo de todo el proceso y se detallan sus ventajas y desventajas.

El propio proceso de desarrollo se subdivide en varias secciones que seguirán el siguiente orden de exposición:

- **Análisis:** en esta sección se explica el modelo de extracción de requisitos y se exponen los mismos.
- **Diseño:** en esta sección se exponen los diferentes prototipos llevados a cabo y se detalla la evolución llevada entre ellos hasta llegar al producto final.
- **Implementación:** en esta sección se muestra la organización de paquetes que tiene el proyecto.
- **Pruebas:** en esta sección se detallan las pruebas a realizar para la validación de la interfaz de usuario y el plan para poder ejecutarlas.

Aunque se muestren como fases secuenciales del desarrollo de proyecto se han ejecutado de forma iterativa obteniendo de cada una de estas iteraciones un nuevo prototipo a mayor detalle y añadiendo requisitos nuevos.

4.1 El proceso de desarrollo

Para poder definir el proceso de desarrollo del proyecto se ha de tener en cuenta que el objetivo del proyecto es ayudar a que un operador de control de la red eléctrica adquiera SA a través de un entorno de visualización. Dicho de otra forma, se trata del diseño de una interfaz preparada para un contexto muy concreto: una sala de control de la red eléctrica.

Por tanto, para alcanzar la solución y los objetivos del proyecto se define un proceso de desarrollo que seguirá el modelo de ciclo de vida de software denominado **Prototipado** [5].

Este modelo se basa en un ciclo de vida en el que las tareas se realizan de manera iterativa. La elección del modelo **Prototipado** se debe a que presenta grandes ventajas de cara al desarrollo y diseño de una interfaz de usuario. Este proceso establece que las tareas del ciclo de vida de software se realicen de forma que se vayan realizando ciclos e ir añadiendo nuevos requisitos o mayor nivel de detalle en los prototipos. La importancia de este modelo es que se incluyen varios prototipos a lo largo del proceso que van permitiendo perfeccionar el producto final. La forma de mejorar a lo largo del proceso es testeando los prototipos que se van obteniendo, así se evoluciona el producto final sin necesidad de que se encuentre disponible. En la [Figura 4](#) se puede observar la conjunción de tareas que conlleva este modelo y cómo se desarrollarían los ciclos o iteraciones.

Al intercalar prototipado entre las tareas de análisis, diseño, implementación y evaluación se consigue dos cosas: saber si la solución diseñada se ajusta a las necesidades del contexto de la solución y que el desarrollador sea más consciente del producto final a alcanzar. Estas dos consecuencias finales desembocan en un desarrollo de implementación más rápido de la solución, puesto que hace el proceso de desarrollo de software más iterativo con retroalimentación por parte de las necesidades del usuario final.

Por otro lado, el proyecto presenta mayor lentitud en la gestión del proyecto ya que, al tratarse de un proceso iterativo, se pueden dar vueltas sobre el diseño hasta que se pongan límites.

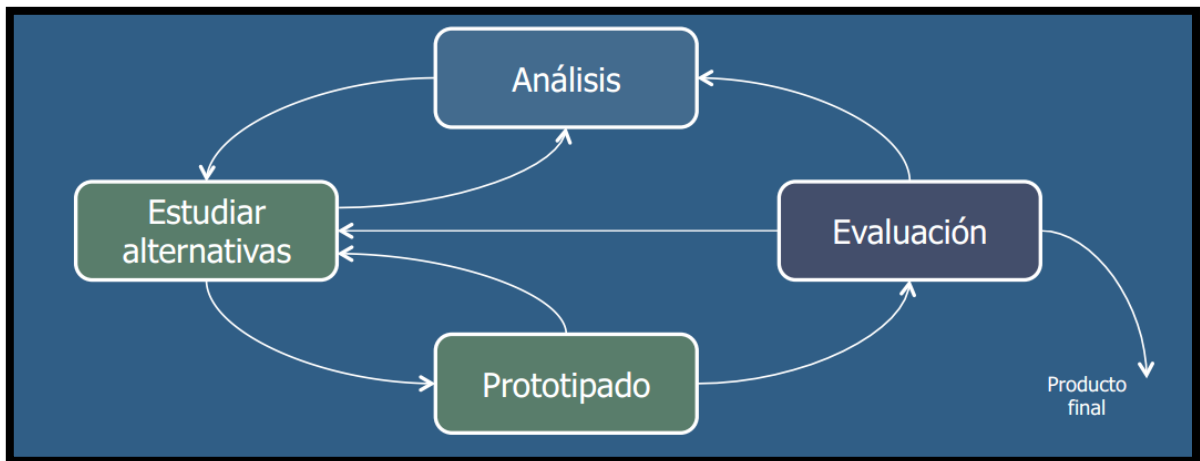


Figura 4: Modelo Prototipado.

A continuación se explican las tareas de: análisis, diseño, implementación y pruebas. Aunque en este documento se muestren de una manera serial es importante recordar que se han realizado hasta 4 ciclos o iteraciones en el proceso de desarrollo. Esto quiere decir que los requisitos mostrados en la etapa de análisis son el cómputo global de las 4 iteraciones. Estos ciclos se explican a mayor detalle en la etapa de diseño, dónde se expone la evolución que conlleva cada una de estas iteraciones como consecuencia de establecer el modelo **Prototipado** como modelo de ciclo de vida de software.

Análisis

El objetivo de esta etapa del proceso de desarrollo es extraer los requisitos para la solución. Para la obtención de los requisitos se ha de tener en cuenta que se trata de una solución para un entorno muy concreto: una sala de control de la red eléctrica. Para poder estudiar el contexto y las necesidades de un operador de la red se propone como método de análisis las entrevistas a expertos. Se ha elegido el método de entrevistas puesto que al tratarse de un entorno muy específico se necesita la opinión de los expertos en el entorno.

Se ha considerado como experto a aquel operador que trabaje en el control de redes eléctricas en la actualidad. Con este método se da paso a que los expertos puedan exponer las carencias del sistema actual e incluso proponer cómo les gustaría trabajar más cómodamente.

Los requisitos obtenidos de estas entrevistas no son todos los que se definen en la siguiente sección, sino que esa lista original se ha ido extendiendo a medida que se han realizado iteraciones en el ciclo de vida de software.

Definición de requisitos

En la fase de análisis se han recopilado los requisitos mínimos para la solución. En esta sección se listan los requisitos recogidos y los requisitos añadidos a raíz de la evaluación de prototipos. La forma de listar los requisitos será siguiendo el siguiente patrón:

- **ID:** En este campo se mostrará el identificador del requisito. El identificador estará compuesto de dos letras definiendo el tipo de requisito. Los requisitos funcionales se identificarán como “FU-XXX”, y los requisitos no funcionales se identificarán como “NF-XXX”. El identificador también llevará tres números que definirán el número de requisito del que se trata.
- **Definición:** En este campo se hará una breve descripción de la necesidad del requisito.

A continuación, se definen los requisitos *funcionales* de la aplicación:

- **ID:** FU-001, **Definición:** La interfaz de usuario deberá comprender cuatro zonas: comunicación, monitorización, control y actuaciones.
- **ID:** FU-002, **Definición:** La interfaz de usuario mostrará una monitorización global de la red eléctrica en la zona de monitorización.
- **ID:** FU-003, **Definición:** La interfaz de usuario permitirá cambiar la vista del mapa espacial de monitorización entre: relieve y callejero.
- **ID:** FU-004, **Definición:** El usuario podrá ampliar cualquier zona del mapa con la función de la lupa.
- **ID:** FU-005, **Definición:** El usuario podrá consultar la información específica de cada alarma.
- **ID:** FU-006, **Definición:** La interfaz de usuario mostrará la información de una alarma en un *tooltip* cerca de la alarma seleccionada.
- **ID:** FU-007, **Definición:** La interfaz de usuario permitirá mostrar el mapa posicional de la red eléctrica.

- **ID:** FU-008, **Definición:** La interfaz de usuario mostrará una vista tabular de datos alfanuméricos.
- **ID:** FU-009, **Definición:** La interfaz de usuario tendrá una coordinación entre la vista tabular y los elementos mostrados en la visualización espacial.
- **ID:** FU-010, **Definición:** En la visualización espacial se ha de mostrar siempre la infraestructura de red.
- **ID:** FU-011, **Definición:** La interfaz de usuario tendrá que mostrar la infraestructura de red con diferentes colores según las zonas de las que se trate.
- **ID:** FU-012, **Definición:** Deberá existir una coordinación entre las 4 zonas de la interfaz de usuario.
- **ID:** FU-013, **Definición:** La interfaz de usuario deberá mostrar en zona de comunicación, la comunicación asíncrona entre los operadores.
- **ID:** FU-014, **Definición:** El usuario podrá enviar mensajes desde la zona de comunicación.
- **ID:** FU-015, **Definición:** El usuario podrá responder mensajes desde la zona de comunicación.
- **ID:** FU-016, **Definición:** El usuario podrá clasificar los distintos mensajes en la zona de comunicación.
- **ID:** FU-017, **Definición:** El usuario podrá seleccionar un área en la visión espacial de la red eléctrica que recoja todas las incidencias de esa zona concreta.
- **ID:** FU-018, **Definición:** El usuario podrá seleccionar una línea en la visión espacial de la red eléctrica que filtre las incidencias de ese tramo.
- **ID:** FU-019, **Definición:** Cada tipo de elemento a mostrar en la visión espacial se albergará en una capa.
- **ID:** FU-020, **Definición:** La interfaz de usuario permitirá al usuario activar o desactivar todas las capas posibles excepto la correspondiente a la infraestructura de la red.
- **ID:** FU-021, **Definición:** La interfaz de usuario marcará las alarmas que surjan nuevas con un parpadeo del icono.
- **ID:** FU-022, **Definición:** La interfaz de usuario marcará las alarmas nuevas en la vista tabular marcando la fila de la alarma con el color de la misma.
- **ID:** FU-023, **Definición:** La interfaz de usuario marcará las alarmas nuevas en la visión espacial con el icono sombreado y redondeado.
- **ID:** FU-024, **Definición:** El color con el que se marcan las nuevas alarmas en la visión espacial perderá intensidad a medida que vaya pasando el tiempo.
- **ID:** FU-025, **Definición:** El usuario podrá acceder a un menú contextual por cada elemento de la infraestructura de red.
- **ID:** FU-026, **Definición:** El menú contextual de cada elemento tendrá las opciones: “Notas”, “Detalle” y “Histórico”.
- **ID:** FU-027, **Definición:** La opción “Histórico” mostrará un gráfico de las medidas eléctricas tomadas en un rango de 10 segundos.

- **ID:** FU-028, **Definición:** La opción “Detalle” mostrará una tabla con cada una de las características del elemento seleccionado.
- **ID:** FU-029, **Definición:** La opción “Notas” mostrará el bloc de notas del elemento seleccionado.
- **ID:** FU-030, **Definición:** La zona de control deberá mostrarse vacía por defecto.
- **ID:** FU-031, **Definición:** Al seleccionar una línea en la visión espacial, en la zona de control se mostrará el mapa unifilar de la línea, un árbol de los componentes de la misma y un listado de incidencias del tramo.
- **ID:** FU-032, **Definición:** Al seleccionar un área en la visión espacial, en la zona de control se mostrará el área seleccionada recortada del mapa, un listado de incidencias y la visión ortogonal del área correspondiente.
- **ID:** FU-033, **Definición:** En la visión espacial deberán estar las capas correspondientes a: brigadas, avisos de clientes, puntos críticos, alarmas e infraestructura.
- **ID:** FU-034, **Definición:** La zona de actuaciones contendrá el listado de actuaciones del operador.
- **ID:** FU-035, **Definición:** El listado de actuaciones estará clasificado por los siguientes tipos de actuaciones: programadas, no programadas, de otro y terminadas.
- **ID:** FU-036, **Definición:** El usuario desplegará una actuación en la zona de control mediante *DragAndDrop*.
- **ID:** FU-037, **Definición:** El usuario podrá recoger la actuación ejecutando un *DragAndDrop* hacia la zona de actuaciones.
- **ID:** FU-038, **Definición:** El usuario podrá quitar todas las ventanas desplegadas en la zona de control con la opción “Limpiar”.
- **ID:** FU-039, **Definición:** Al seleccionar un área o una línea, se podrá crear una actuación mediante la opción “Crear” en la zona de control.
- **ID:** FU-040, **Definición:** Al crear una actuación será del tipo no programada.
- **ID:** FU-041, **Definición:** Al crear una actuación se iniciará un *log* de registros sobre esta, de las actividades realizadas en la actuación.
- **ID:** FU-042, **Definición:** Las actuaciones programadas serán identificadas con el color azul.
- **ID:** FU-043, **Definición:** Las actuaciones no programadas serán identificadas con el color rojo.
- **ID:** FU-044, **Definición:** Las actuaciones de otro serán identificadas con el color amarillo.
- **ID:** FU-045, **Definición:** Las actuaciones terminadas serán identificadas con el color verde.
- **ID:** FU-046, **Definición:** Al seleccionar un área, se podrán activar o desactivar las capas de elementos en el recorte del área que se encuentra en la zona de control.
- **ID:** FU-047, **Definición:** La interfaz de usuario permitirá mostrar la secuencia de aparición de alarmas desde el momento actual hasta 4 minutos atrás.

- **ID:** FU-048, **Definición:** La interfaz de usuario permitirá recrear una simulación de las alarmas que puedan aparecer desde el momento actual hasta 4 minutos en el futuro.
- **ID:** FU-049, **Definición:** La interfaz de usuario deberá indicar al usuario cuando está visualizando una simulación.
- **ID:** FU-050, **Definición:** El usuario podrá editar el *log* de registros de una actuación.

En el siguiente listado se definen los requisitos *no funcionales* de la aplicación:

- **ID:** NF-001, **Definición:** El sistema será ejecutado en un monitor curvo de 42".
- **ID:** NF-002, **Definición:** El sistema será ejecutado sobre el sistema operativo Windows.
- **ID:** NF-003, **Definición:** El sistema deberá actualizar la información cada 0,5 segundos.
- **ID:** NF-004, **Definición:** La interfaz de usuario seguirá el estándar **Abnormal Situation Management** (ASM) [6] para la clasificación de alarmas.
- **ID:** NF-005, **Definición:** El sistema deberá ser una aplicación de escritorio.
- **ID:** NF-006, **Definición:** El entorno de ejecución deberá tener instalado Microsoft Office 2010.

Especificación de requisitos

En esta sección se detallan cada uno de los requisitos definidos en la sección anterior. Para llevar a cabo una especificación correcta de los requisitos se seguirá la plantilla que plantea Volere [7] para detallar los requisitos.

Esta plantilla contiene diferentes campos que no van a ser utilizados para la especificación de requisitos de este proyecto por las razones que se exponen a continuación:

- **Casos de uso:** no se especifican casos de uso por lo que no tiene coherencia relacionar requisitos con casos de uso.
- **Razón:** la justificación de todos los requisitos es trivial, ya sea por el contexto en el que se quiere implantar la interfaz de usuario o por las necesidades del sistema.
- **Origen:** el origen de todos los requisitos reside en el diseño de la interfaz, por lo que no tiene sentido añadir este campo siendo el mismo para todos.
- **Satisfacción del cliente:** los requisitos son evaluados por el diseñador y el investigador del proyecto por lo que el cliente no tiene cabida.
- **Conflictos:** se trata de requisitos para una interfaz de usuario luego no existen conflictos entre ellos.
- **Material:** al ser diseño de una interfaz de usuario no es necesario variar de materiales para cumplir cada requisitos, sino que todos se implementan en el mismo software.
- **Historial:** Puesto que son requisitos originados de un contexto o una necesidad del sistema no van a sufrir cambios, luego un histórico del requisito no es necesario.

Con la plantilla de Volere adaptada a las necesidades del proyecto queda una ficha para la especificación de requisitos como la que se muestra a continuación.

Identificador			
Tipo de Requisito		Prioridad	
Descripción			
Prueba de Verificación			

Tabla 10: Plantilla requisitos.

La plantilla que se ha definido se compone de cinco campos. El significado de cada uno de los campos a rellenar en la especificación de requisitos es el explicado a continuación:

- **Identificador:** este campo mostrará el identificador del requisito. El campo estará compuesto de dos letras definiendo el tipo de requisito (funcional o no funcional). Los requisitos funcionales se identificarán como “FU-XXX”, y los requisitos no funcionales se identificarán como “NF-XXX”. Este campo llevará, a continuación de las letras, tres números que definirán el número que identifica el requisito. Este campo coincide con el campo **ID** del listado de la sección anterior, [Definición de requisitos](#).
- **Tipo de Requisito:** este campo definirá el tipo del requisito, siendo este tipo un nivel más de profundidad en la clasificación de requisitos. Los requisitos se dividen en *funcionales* y *no funcionales*, esta división viene dada en el campo **Identificador**. Dentro de estas dos divisiones se pueden clasificar, según a quién afecte el requisito, al *usuario* o al *sistema*.
- **Prioridad:** este campo establece el grado de importancia de un requisito frente al resto según los objetivos del proyecto. Los valores que puede tomar son:
 - **Alta:** si el requisito es imprescindible para conseguir los objetivos del proyecto.
 - **Media:** si el requisito es necesario pero no primordial para el desarrollo del proyecto.
 - **Baja:** si el requisito no influye en el desarrollo del proyecto.
- **Descripción:** este campo contemplará una breve descripción de la necesidad que plantea el requisito para el proyecto.
- **Pruebas de Verificación:** este campo contendrá una descripción de la prueba que se ha de llevar a cabo para comprobar si la solución cumple el requisito o no.

La especificación de requisitos se desarrolla en el [Anexo III. Especificación de requisitos](#).

Diseño

En esta etapa del proceso de desarrollo se ha dividido en el diseño en cuatro fases que vienen delimitadas por diferentes niveles de detalle de los prototipos generados. Cada fase se identifica con una iteración en el proceso de desarrollo de software. A pesar de que se producen varias iteraciones en el ciclo de vida del software en todas ellas se ha diseñado para un monitor curvo de

43 pulgadas. En esta etapa de diseño se detalla el seguimiento del modelo de proceso de vida de software propuesto en el apartado [El proceso de desarrollo](#) : modelo de **Prototipado**.

Las cuatro fases en las que se divide la etapa de diseño son las que se definen a continuación, siguiendo el mismo orden en el que se ejecutaron en el desarrollo:

- **Fase 1:** En esta fase se llevará a cabo un primer mockup de la interfaz con los resultados obtenidos de la fase de análisis.
- **Fase 2:** En esta fase se realizará dos prototipos básicos de la interfaz en las tecnologías: Java RCP y *Windows Presentation Foundation*.
- **Fase 3:** En esta fase, y tras haber elegido la tecnología adecuada, se desarrollará a mayor nivel el prototipo.
- **Fase 4 Producto final:** En la última fase, se proponen las últimas mejoras posibles en la interfaz diseñada y la obtención del producto final.

A continuación, se detalla el proceso y resultado de cada una de las fases en las que se divide la etapa de diseño. Se detallan en el mismo orden en el que se llevan a cabo puesto que la salida de cada una de las fases es la entrada que recibe la siguiente fase.

Fase 1

Esta fase comienza con el análisis de los requisitos obtenidos en las entrevistas a los expertos y el estudio del contexto. El objetivo de esta fase es conceptualizar la solución. Esto quiere decir, que el prototipo que se lleva a cabo en esta fase tiene como principal objetivo contemplar los conceptos básicos necesarios para asistir a un operador en la adquisición de SA. Por lo tanto, con este prototipo se busca definir el *layout* de la interfaz de usuario. De esta forma, se podrá llevar a cabo una evaluación de la interfaz de usuario resultante del análisis de las necesidades del operador.

En esta fase se realiza un prototipo del tipo horizontal. Estos prototipos son aquellos que se basan en mostrar las características principales de la interfaz de usuario sin tener un respaldo de la funcionalidad. El prototipo llevado a cabo es de baja fidelidad puesto que sólo determina zonas e instrumentos para la realización de tareas básicas. Además, la interfaz está planteada con un nivel de detalle muy bajo, mostrando únicamente un mero indicio de los conceptos del planteamiento de la solución. Este prototipo se engloba en los denominados *Wireframe*, que comprenden los prototipos horizontales y de bajo nivel que pueden ser evaluados por el usuario.

Para realizar este prototipo son necesarias herramientas software para el diseño de prototipos de bajo nivel de fidelidad. En este caso, se ha utilizado la herramienta Pencil [8].

El prototipo diseñado en esta fase parte de los conceptos principales obtenido tras las entrevistas con expertos. Estos conceptos coinciden con los elementos imprescindibles de un operador para el control de la red eléctrica. El principal trabajo del operador parte de que éste ha de tener monitorizada y controlada la red. Cada uno de estos conceptos conlleva una alta importancia en el desarrollo del trabajo de un operador.

En el caso de la monitorización de la red, el operador necesita una visión espacial completa de la zona de red eléctrica en la que trabaja. Es en esta área, donde se pretende mantener al

operador informado del estado de la red y de las posibles emergencias que puedan surgir. Esta visión global de todo lo que ocurre en la red eléctrica es uno de los principales pasos necesarios para que el operador adquiera el SA. A la par que la monitorización global que debe recibir el operador, se ha de definir una zona donde éste pueda controlar un suceso de forma concreta. Esta necesidad es la que impone que exista una zona dirigida al control de sucesos o líneas. Estos dos conceptos son los que permiten la realización del operador sin que éste pierda conocimiento de la situación o estado de la red eléctrica global. Un tercer concepto a representar e incluir en el prototipo de esta fase, es el área de proyección. Esta área incluiría una serie de representaciones tanto alfanuméricas como espaciales en las cuales se podría observar la progresión de la red eléctrica en el momento actual o en un futuro inmediato. El último concepto clave para la definición del *layout* de la interfaz de usuario, se corresponde con la comunicación y externalización de los sucesos u operaciones que se han de llevar a cabo en la red eléctrica.

Luego, el diseño del *layout* está obligado a tener estos cuatro conceptos de forma visual con la condición de que no se puedan superponer u ocultar entre ellos. Esta condición viene dada por la adquisición del SA, el operador no debe perder conocimiento en ningún proceso del resto de elementos necesarios para desarrollar su trabajo. Como se puede ver en la [Figura 5](#), esta necesidad descrita se traduce en la generación de cuatro zonas concretas y bien diferenciadas en la interfaz de usuario.

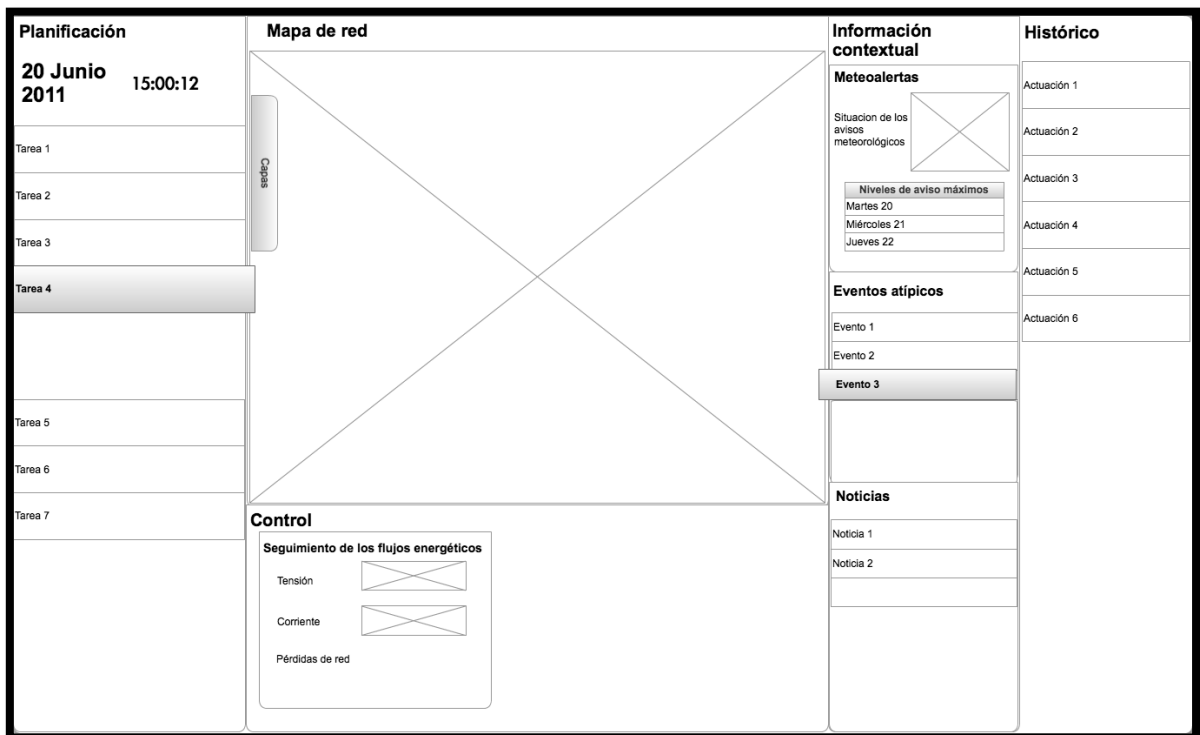


Figura 5: Prototipo 1.

La conclusión principal y más importante de este prototipo es la afirmación de que se pueda integrar en un sistema de control todas las herramientas que utilizan los operadores en su labor. Una segunda conclusión es la desaparición del área de proyección de cara a la siguiente fase. Esta decisión se toma puesto que el contenido y objetivo se corresponde con herramientas que ofrecen la zona de monitorización y la de control. Por lo tanto, el área de proyección se ve embebida en las dos zonas especificadas. Sin embargo, el otro objetivo de éste área relacionado con la proyección de un futuro inmediato a modo de “predicción” sencillamente desaparece del *layout* de la interfaz de usuario.

Fase 2

En la segunda fase del diseño, y la segunda iteración en el ciclo de desarrollo de software, se tiene por objetivo establecer los componentes de la interfaz y los nodos de información específicos. Para poder llevar a cabo este objetivo, se parte del *layout* ya definido tras la evaluación del primer prototipo. Por lo tanto, en esta fase se busca detallar un nivel más la forma de visualizar toda la información que requiere un operador. Puesto que esta fase requiere la selección de diferentes componentes visuales, es en esta fase cuando se realiza una comparación de tecnologías a nivel experimental.

A diferencia de la fase anterior, el prototipo a desarrollar en ésta fase se trata de un prototipo software. Para el desarrollo del prototipo software se utilizan las tecnologías: *Windows Presentation Foundation* y *Java Rich Client Platform*. Para desarrollar en cada una de estas tecnologías son necesarias herramientas software que ofrezcan el entorno de desarrollo apropiado. Estas herramientas son: para la tecnología Java RCP es necesario el software Eclipse [9], y para la tecnología WPF es necesario el software Visual Studio 2012 [10]. Con estas tecnologías se consigue un prototipo de alta fidelidad que contiene un mayor nivel de detalle, además de la disposición funcional del *layout*. El prototipo llevado a cabo en esta fase comienza a tener un aspecto funcional de la interfaz pero sigue predominando la exposición de los componentes visuales y las características del diseño, por lo que se trata de un prototipo más horizontal que vertical.

Aunque se ha de realizar el prototipo en dos tecnologías se comprende el mismo alcance para ambas. El prototipo de esta fase muestra la disposición final de las zonas obtenida en el prototipo de la fase 1. La disposición se centra en tres zonas bien diferenciadas, además, en este prototipo se ha de concretar el contenido en forma de componentes visuales que ha de tener cada una de estas zonas. La colocación de las zonas en ambos prototipos es la misma, de izquierda a derecha: zona de comunicación, zona de monitorización, zona de control y zona de comunicación. Esta colocación se debe a intentar procurar mayor importancia a las dos zonas centrales, por lo tanto, se divide la zona de comunicación en dos espacios en la interfaz para que ocupen el menor espacio posible en la visión del operador. A continuación se detalla dicho contenido por cada una de las zonas:

- **Zona de monitorización:** La monitorización de la red implica realizar un seguimiento de la misma lo que significa que el operador sea consciente del estado de la infraestructura, los eventos críticos, situaciones anormales, etc. Toda esta información

se representa mediante dos vistas gráficas y una vista tabular. En la primera vista gráfica se muestra desplegada la información sobre la conectividad de la red según la posición geográfica. Esta primera vista gráfica se denominará **vista posicional**. La segunda vista gráfica muestra un gráfico de barras paralelas que recogen los diferentes valores de la red eléctrica. De tal forma que se otorga al operador una visión general de los eventos e incidentes críticos agrupados por hora, tipo y dispositivos eléctricos. Por último, la vista tabular muestra una lista vertical de alarmas reportadas por la red. Esta vista ofrece diferentes herramientas para la clasificación de los elementos como: filtrado, ocultación, buscar, etc. Las tres vistas descritas para componer la zona de monitorización estarán coordinadas entre sí, de manera que se pueda analizar un elemento de la red desde diferentes puntos de vista sin que se tenga que ocultar ninguno de ellos.

- **Zona de control:** El control de la red implica conocer el conjunto de los acontecimientos y los atributos de conectividad de un área determinada de la red eléctrica. El motivo de esta “filtración” de información por un área específico es asistir a las operaciones de bajo nivel que pueda realizar el operador. En esta zona se comprenden dos vistas gráficas esquemáticas. La primera vista gráfica se corresponde con un diagrama de la línea seleccionada. Esta vista se centra en la conectividad de la red pero no ofrece distancias geográficas reales entre los dispositivos incluidos en la línea. La segunda vista se corresponde con un diagrama topológico y temporal del área seleccionada. Esta vista muestra los eventos de la red en relación temporal permitiendo al operador identificar correlaciones y tendencias. Es en esta vista donde se aplican las herramientas que proponía el área de proyección (área disuelta entre la zona de monitorización y control) del primer prototipo, ya que el operador puede identificar patrones para el apoyo a la proyección de la situación.
- **Zonas de comunicación:** El objetivo de estas zonas es ayudar al operador a mantener una comunicación síncrona y asíncrona con otros operadores o con personal de campo. Esta área se divide en los prototipos en dos zonas una situada en el lado derecho de la interfaz y otra en lado izquierdo. La zona del lado derecho se centra en permitir al operado crear, filtrar y priorizar la información relacionada con los eventos importantes ocurridos en la red. La zona del lado izquierdo se centra en ofrecer al operador una herramienta para que pueda comunicarse a tiempo real con el resto del personal.

A continuación, conociendo ya el alcance del prototipado en esta fase, se van a exponer los prototipos generados con las tecnologías que se comparan. Para la comparación experimental de las tecnologías se ha referenciado a los criterios definidos en el apartado [Criterios de evaluación](#).

El prototipo obtenido con Java *Rich Client Platform* es el que se muestra en la [Figura 6](#). El *layout* que muestra este prototipo varía en pequeños detalles del definido en el primer prototipo. Esto es debido a que la tecnología no permitía definir una estructura de ventanas al desarrollador sino que se debían de cumplir ciertas restricciones impuestas por la tecnología.

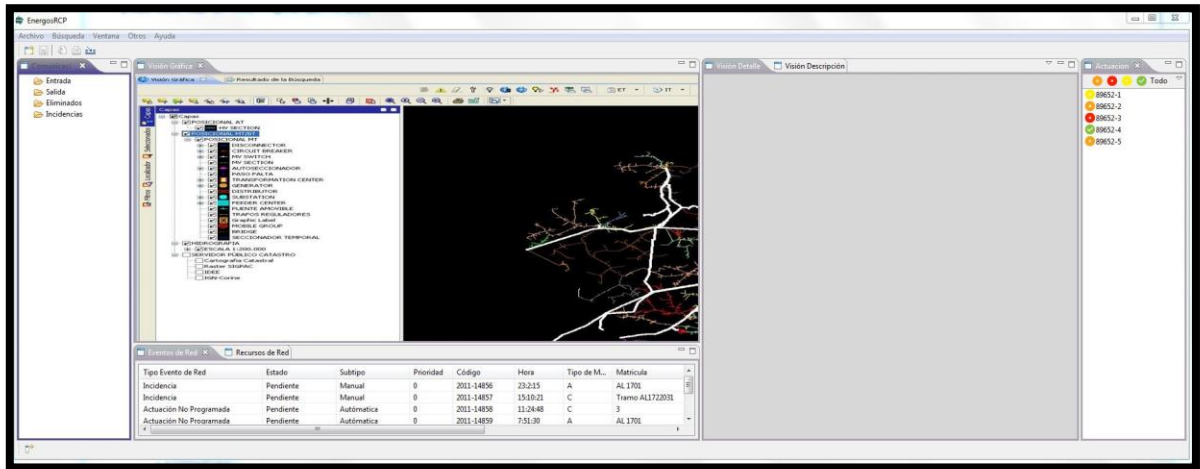


Figura 6: Prototipo Java Rich Client Platform.

La principal restricción de la tecnología es que el usuario sólo podrá visualizar una **perspectiva**. Las perspectivas en Java RCP definen el contenedor de visual de la aplicación. Dentro de cada perspectiva se pueden incluir varias **vistas**. En la Figura 6, se pueden identificar las vistas generadas en la perspectiva como cada uno de los contenedores cuadrados e independientes. Otro aspecto condicionante de la tecnología, es que cada una de las vistas que se incluyan en la perspectiva se colocan siguiendo un patrón similar a una pila. Cada vista se va acoplando a la anterior por orden de inicialización y sólo se les puede otorgar un atributo que indique la posición de la pantalla donde tienen prioridad. Este atributo sólo acoge los valores: *Right*, *Left*, *Top* o *Bottom*. Una característica importante a la hora de comparar las tecnologías es que en ésta no se permite incluir vistas dentro de vistas. Por lo tanto, sólo se permite una estructura estática con dos niveles de profundidad: una perspectiva compuesta de varias vistas. Estas restricciones de la tecnología ya condicionan en gran medida la disposición y gestión del *layout* definido en el primer prototipo. La interacción del usuario a la hora de utilizar el prototipo sólo se puede realizar mediante botones clásicos y *DragAndDrop* con elementos muy concretos. Todas estas condiciones limitan en gran cantidad a esta tecnología en el criterio de comparación **interacción rica en la interfaz**. A la estructura fija de los contenedores se le suma la condición de no poder incluir en una vista diferentes componentes visuales, por ejemplo una imagen y una tabla. Por cada vista sólo se permite definir un tipo de componente, exceptuando la inclusión de botones. Partiendo de la poca libertad para compaginar componentes hay que destacar la imposibilidad de generar o modificar componentes visuales. Sólo se pueden utilizar los componentes clásicos sin variación como: tabla alfanumérica, lista de objetos, árbol de carpetas visual, etc. Por lo tanto, se trata de una tecnología muy pobre en el criterio de comparación **componentes visuales**.

El desarrollo de este prototipo llevó más tiempo de lo esperado puesto que existen escasas referencias que se puedan consultar para el aprendizaje de la tecnología. Además, la documentación encontrada, únicamente en formato web, sólo ofrece solución a problemas básicos y sin fundamento para la composición de una buena interfaz. Esta falta de material de

apoyo hace que esta tecnología tenga un valor muy bajo en el criterio de comparación **material de ayuda y referencias**.

El siguiente prototipo a detallar es el correspondiente con la tecnología *Windows Presentation Foundation*.

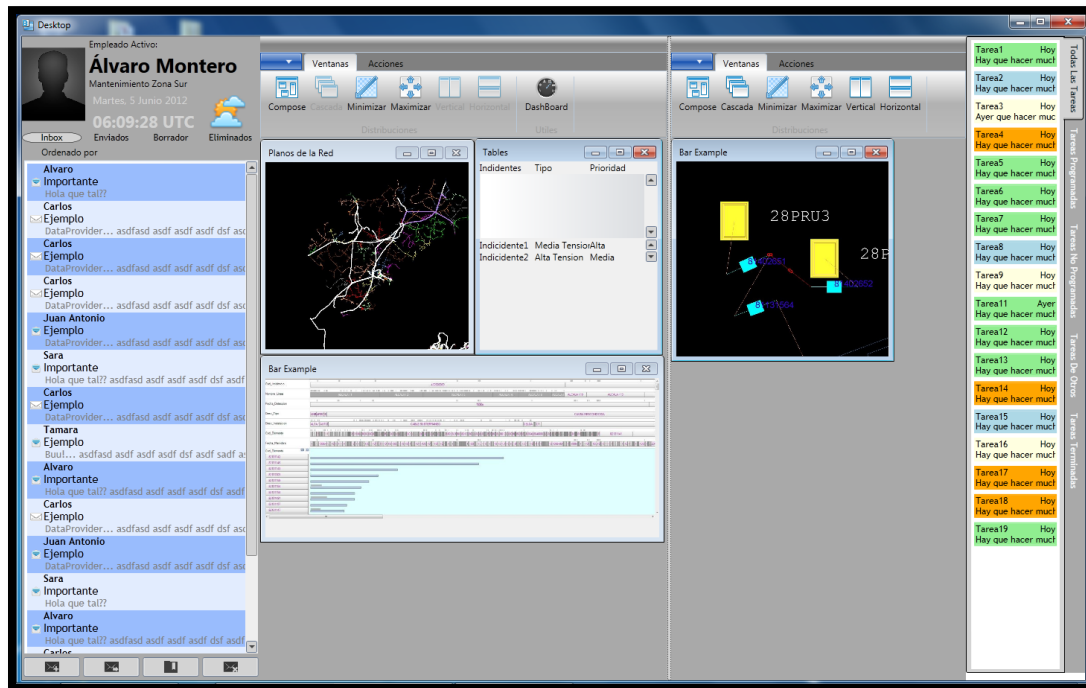


Figura 7: Prototipo Windows Presentation Foundation.

El prototipo obtenido mediante la utilización de la tecnología WPF resulta tener gran potencia en los criterios de comparación en los que se ha evaluado el prototipo de Java RCP. En primer lugar, en el prototipo destaca la definición de las cuatro zonas necesarias. En este caso cada zona se distingue de las otras ya que a cada una de ellas se le ha otorgado un bloque a modo de columna en la pantalla. Cada uno de estas vistas contiene a su vez diferentes vistas. Para la interacción con las vistas internas de cada zona se encuentran diferentes posibilidades: listado de objetos, *tab panel*, desplegables, movilidad de cada vista, redimensión de cada vista, diferentes tipos de visualización de ventanas, etc. Con ello se demuestra que esta tecnología tiene una gran potencia en los criterios de **interacción rica en la interfaz**. Ligado a este criterio va el referente a los **componentes visuales**. En el caso de esta tecnología se puede observar que la manera de mostrar la información en el prototipo es muy variada, permitiendo desde la conjunción de imágenes, tablas y texto en una misma zona hasta la creación de contenedores específicos para un objeto concreto dentro de la aplicación. Un ejemplo de esto último sería la visualización de la bandeja de entrada del correo interno y la visualización de la lista de tareas o actuaciones que ha de realizar el operador. Por lo tanto, WPF en este criterio supera a Java RCP dando una mayor libertad al diseñador y al desarrollador.

El desarrollo de este prototipo llevó menos tiempo del que se presupone al observar el prototipo. Esto fue debido a que existe una gran cantidad de referencias, tanto de formato web como formato libro. Por ello, esta tecnología supera en la anterior en el criterio **material de ayuda y referencias**.

Tras la evaluación de los prototipos presentados a los expertos se obtienen tres conclusiones importantes a tener en cuenta para la entrada de la siguiente fase. La primera conclusión es la necesidad de incluir una vista geográfica de la red. Esta vista apoyaría a la visión general del estado de la red eléctrica. La segunda conclusión es el cambio de las representaciones de información detallada. El operador requiere una serie de valores numéricos constantemente para poder supervisar la red eléctrica y la representación tomada en los prototipos no era lo suficientemente ágil para que el operador pudiera trabajar con ello. La tercera, y última, conclusión es la separación por semántica de las zonas de comunicación. Se divide en dos zonas diferentes: **zona de comunicación** correspondiente a la comunicación asíncrona del operador con el resto del personal; **zona de actuaciones** correspondiente a la clasificación de la información relacionada con los eventos ocurridos en la red.

Fase 3

Tras la segunda iteración sobre el diseño del prototipo se tiene como conclusiones la colocación de las zonas en la pantalla y los componentes necesarios para la visualización de la información. Esta es la entrada para la siguiente iteración en el ciclo de vida del software, la tercera fase. El objetivo de esta fase reside en la modulación de la interacción del operador y la visualización de información de la red.

Al igual que en la fase anterior, en esta se desarrolla un prototipo software. Aunque se incluye la mínima funcionalidad para hacer posible la ejecución y representación del escenario, se trata de un prototipo horizontal. Esto es debido a que, el objetivo de este sigue siendo la evaluación del diseño de la interfaz de usuario y la interacción de la misma. Tras la comparación experimental realizada en la segunda fase se decide optar por la tecnología *Windows Presentation Foundation* como tecnología óptima para el desarrollo del prototipo software. Por lo tanto, el entorno de desarrollo es el software Visual Studio 2012.

Partiendo de la salida obtenida de la segunda fase, en esta iteración se busca alcanzar una interfaz más amigable de cara al operador. Incluyendo en esta interfaz las necesidades obtenidas de la evaluación del anterior prototipo como la colocación de un mapa para dar una visión espacial y poder, así, ampliar el conocimiento del operador sobre la red en la zona de monitorización.

En este prototipo se busca incluir nuevas posibilidades de visualización y mejoras en los componentes de la interfaz. A todo ello se le suma la necesidad de incluir nuevas formas de interacción entre el operador y la interfaz, entre ellas el *DragAndDrop*, el dibujo de un área definida por el operador, *MouseOver* para la visualización de información, etc.

Otra necesidad nueva es la inclusión de la zona de actuaciones dentro de la zona de control para dar mayor espacio a la monitorización y control de la red. De tal forma que la disposición final de las zonas definidas es la expuesta en la [Figura 8](#).

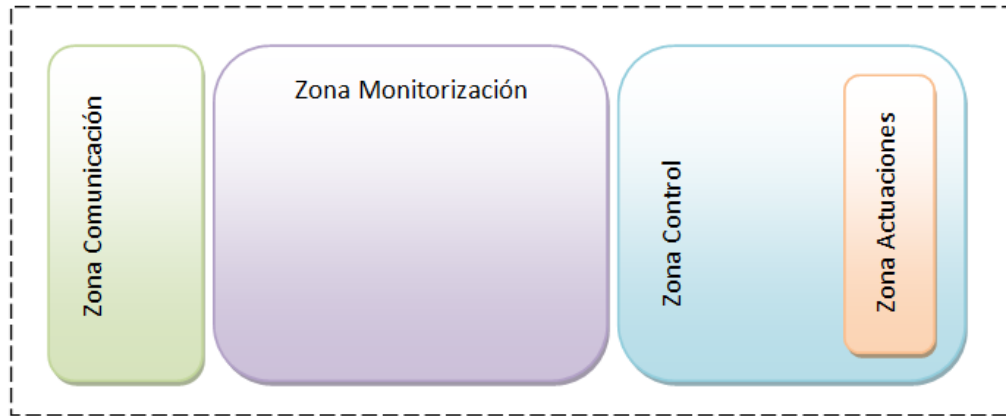


Figura 8: Disposición final de zonas.

En el segundo prototipo desarrollado con la tecnología WPF se ha modificado la apariencia y la colocación de las zonas. En cuanto a los menús de cada zona y la apariencia de ventanas y componentes se ha utilizado como base el estilo de *Microsoft Aero Glass*, siendo modificado en cuanto al color de fondo.

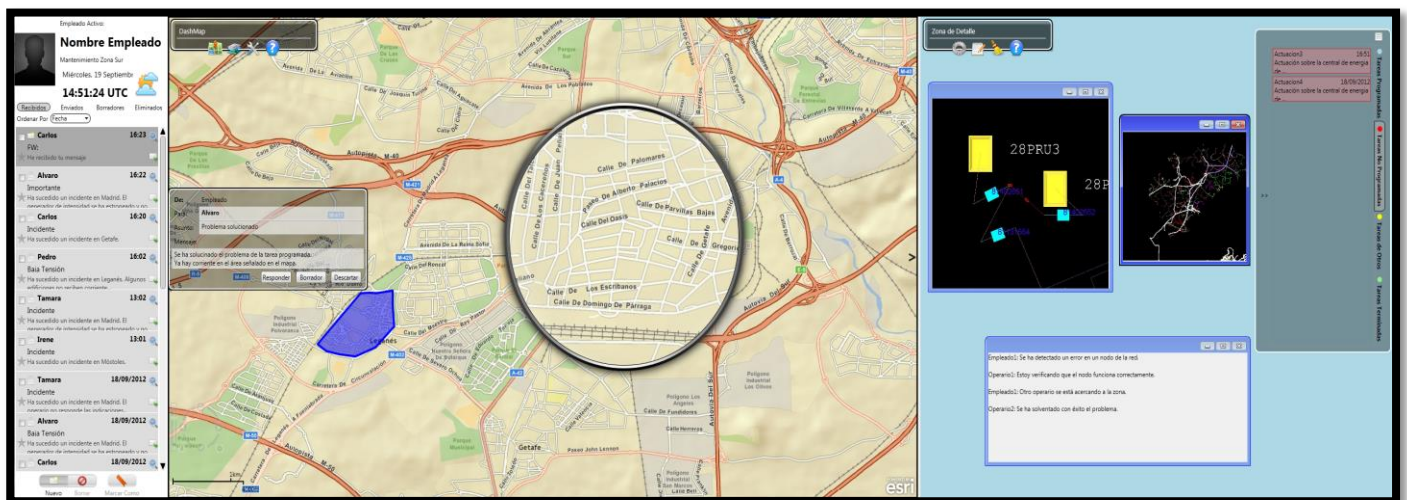


Figura 9: Segundo prototipo WPF.

Con el objetivo de aumentar el espacio para la zona de control se ha optado por generar un componente visual, determinará la zona de actuaciones, semitransparente que contenga el listado de actuaciones. Dentro de este panel transparente se encuentra la clasificación y filtrado de las actuaciones mediante un *tab panel* que despliega la lista de actuaciones del tipo seleccionado. Este componente puede cambiar de geometría pasando a tener las actuaciones simulando una pila, quedando como se muestra en la Figura 10.

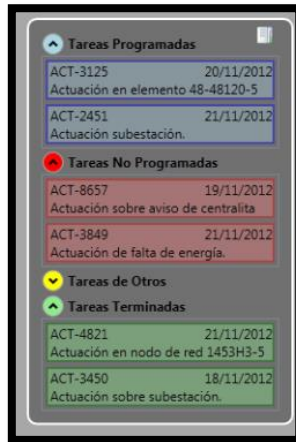


Figura 10: Pila actuaciones.

Al incluir esta zona en la zona de control, el despliegue de cada actuación se realiza en la zona de control. Por lo tanto, la zona de actuaciones únicamente actúa como un listado de las mismas. Para poder desplegar u ocultar una actuación desplegada se ha de realizar un *DragAndDrop* desde una zona a la otra. En la zona de control, se incluyen diferentes ventanas según los datos que contengan la actuación o el área seleccionada de la visión espacial. Estas ventanas permiten ser movidas de manera libre por toda la zona, redimensionadas, minimizadas, maximizadas o cerradas.

En la zona de monitorización se ha incluido un mapa con visibilidad de callejero por defecto y que el operador podrá cambiar a satélite desde el menú principal de la zona en cualquier momento. La necesidad de este tipo de mapas se basa en que la *Smart Grid* no sólo toma datos eléctricos sino que también toma datos a nivel de usuarios de red. Así, con un mapa geográfico se puede monitorizar todas las zonas eléctricas que se encuentran relacionadas entre sí. El mapa incluido es tomado de la librería de ArcGIS [11]. Se ha seleccionado esta librería puesto que permite la utilización de capas de información sobre los mapas. De esta forma, por cada capa, en el mapa se mostrarán los elementos correspondientes. La inserción de cada uno de los elementos se realiza mediante coordenadas. En esta zona también se han generado diferentes herramientas: una lupa para ampliar el detalle de la red, selección de área, activación de capas, etc. La visión espacial de la red se implementa mediante capas en el mapa mostrado. Esto quiere decir, que cada capa mostrará un tipo de elemento de la red eléctrica. Así se tendrán como capas: la infraestructura de red, las alarmas, puntos críticos, avisos de clientes y las brigadas activas. El operador podrá activar y visualizar las capas que desee mediante el menú principal de la zona de monitorización. La selección de área es una herramienta para que el operador pueda dibujar un polígono sobre el mapa, de tal manera que la zona seleccionada será filtrada y desplegada en la zona de control con mayor información acerca de ella. Por último, en la zona de comunicación se ha modificado la estética de la bandeja de entrada adaptando los colores al tema general de la interfaz de usuario. En esta zona también se han incluido dos componentes que permiten la visualización y respuesta de correos al posicionar el ratón sobre el icono debido. El panel para

estas dos acciones se despliega al lateral de la zona de comunicación posicionándose sobre el mapa de la zona de monitorización, como se muestra en la [Figura 9](#). Los correos de la bandeja de entrada se pueden filtrar mediante un *radio button* modificado que se encuentra en la parte superior de la lista.

Para cerrar la iteración del ciclo de vida del software se plantea una evaluación del prototipo obtenido en esta fase por parte de los expertos. De esta evaluación se extraen las conclusiones y mejoras que se exponen a continuación. Es en esta evaluación dónde más información y conclusiones se consiguen extraer puesto que los expertos pueden simular de una forma más realista el funcionamiento de la interfaz de usuario y, por lo tanto, pueden exponer con mayor detalle los problemas encontrados o las carencias. Para identificar mejor cada una de las mejoras se explican a continuación clasificadas por la zona a la que pertenecen:

- **Zona de control:** La representación de la información no se corresponde con la información que requieren los operadores al tratar un área de la red. Esto conlleva que se ha de plantear otra serie de datos a mostrar entre ellos: la visión ortogonal del área seleccionada, el filtrado de incidencias en la misma, etc. Cuando el operador se encuentre con un área del mapa filtrada tiene que existir la posibilidad de generar una nueva actuación con esos datos. Dentro de cada actuación se ha de tener un *log* de registros de las acciones que realiza cada operador que pueda servir para la comunicación con otros compañeros. En caso de que el operador no quiera generar una nueva actuación tiene que existir la posibilidad de limpiar la zona de actuación devolviéndola al modo por defecto.
- **Zona de monitorización:** La capa de infraestructura de la red tiene que ser fija, el operador no debe poder desactivarla. Para apoyar la monitorización de la red, el operador necesita la visión posicional de la red global y los datos alfanuméricos que informen detalladamente de los elementos de red. El operador tiene que poder seleccionar un área, a definir por él, o seleccionar una única línea de la red.

Fase 4: Producto Final

En la tercera iteración del ciclo de vida del software se ha obtenido un prototipo software de alta fidelidad en cuanto a la interfaz de usuario. Sin embargo, como ya se ha planteado, tras la evaluación surgieron nuevas necesidades y carencias. Por lo tanto, en la cuarta fase se tiene por objetivo solventar todos los problemas expuestos por los expertos y añadir detalles o componentes que se consideren necesarios desde el punto de vista del diseñador.

Para poder alcanzar estos objetivos se realiza un prototipo software de alta fidelidad. Este prototipo se lleva a cabo sobre la tecnología WPF y, a pesar de que se incluyen ciertas funcionalidades como la conexión a base de datos o algoritmos de filtrado del área, se considera un prototipo horizontal puesto que todos los objetivos de desarrollo son acerca del diseño de la interfaz de usuario. Al desarrollarse sobre la misma tecnología que los anteriores, el entorno de desarrollo sigue siendo el mismo el software Visual Studio 2012.

El prototipo desarrollado en esta cuarta fase resulta ser el producto final, es decir, el diseño final de la interfaz de usuario para apoyar a que el operador adquiera el SA. A lo largo de esta

sección se describe cada una de las modificaciones o de los componentes insertados nuevos en el prototipo a diferencia del anterior prototipo obtenido.

En la zona de monitorización destacan varias modificaciones. La visión espacial tanto en el modo callejero como relieve muestra siempre el mapa de la infraestructura de red. Esta infraestructura ha variado de manera que se distinguen las zonas definidas por la red mediante la agrupación de líneas por colores. En la infraestructura existe una estación por cada zona y y subestaciones de unión. En la visión espacial se representa la estación de color azul y las subestaciones de color gris. A estas visiones se le suma la inclusión de un nuevo icono en el menú que permite al operador cambiar a una visión del posicional de la red, como se muestra en la Figura 11.

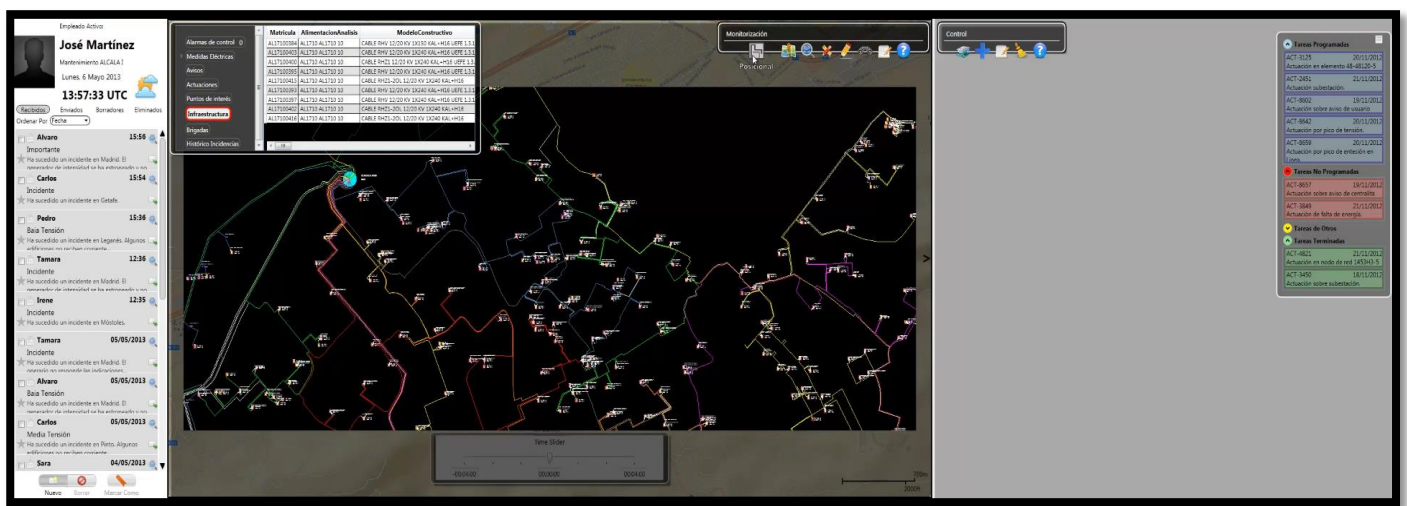


Figura 11: Visión posicional de la red.

En la visión posicional no se permite la inserción de capas puesto que esta ya ofrece una visión muy detallada de las líneas de la red eléctrica. Sin embargo, en la visión espacial, tanto en modo callejero como relieve, resulta necesaria la aplicación de capas para que el operador pueda filtrar la información según las necesidades de la red. A esta necesidad se le suma que el operador no sólo necesita información visual o gráfica sino también contenido alfanumérico que pueda ayudar a complementar la información que se recibe desde la visión espacial. La solución llevada a cabo es la creación de una **visión tabular** o tabla, que albergue los datos correspondientes cada elemento que se encuentre activo en la red. Dentro de esta tabla se encuentra un árbol de tipos o capas que clasifica los elementos agrupándolos. Este árbol también contiene etiquetas de grupos de elementos que no son visibles en la visión espacial: histórico de incidencias y medidas eléctricas. El operador podrá activar y desactivar cada capa a medida que consulte los datos alfanuméricos. De esta forma, se consigue que la visión tabular esté coordinada con la visión espacial. Cuando el operador activa o desactiva cada capa de información, los elementos del mapa son recogidos por su posición en coordenadas y son relacionados con los objetos de la base de datos. Cada capa que el operador active se mostrará en la visión espacial y se indicará en el árbol de la vista tabular

cambiado su etiqueta a un sombreado blanco. En caso de que se hayan activado varias capas, para poder diferenciar a cuál corresponden los datos alfanuméricos desplegados en la vista tabular se mostrará un borde rojo alrededor de la etiqueta del grupo de elementos correspondiente como se muestra en la [Figura 12](#). Las características que presenta la tabla donde se despliegan los datos son las características básicas de cualquier tabla, permite el *scroll* en horizontal y vertical y la selección de elementos. Si el operador selecciona elementos de los datos de una de las capas activas, en la visión espacial se identificará el elemento seleccionado. Para cada tipo de elemento se identificará de una manera distinta:

- Los avisos a clientes, al ser seleccionados, cambiarán de un icono gris a un icono azul.
- Las brigadas que estén operando, al ser seleccionadas, en la red cambiarán de un icono rojo a un icono naranja.
- Las estaciones y subestaciones de la infraestructura de red, al ser seleccionadas, agrandarán el mismo icono y se les aplicará un borde negro alrededor que refuerce la llamada de atención al operador.
- Las alarmas, al ser seleccionadas, se le añadirá al icono una mirilla cuadrada en color negro que permitan al operador identificarlas de manera rápida.

En la vista tabular es donde se requiere un acceso a datos externos. Se ha implementado dos tipos de accesos: a base de datos en lenguaje MySQL [12] y a las hojas de cálculo Excel [13]. La información de la vista tabular se actualizará a la par que la información de la visión espacial.

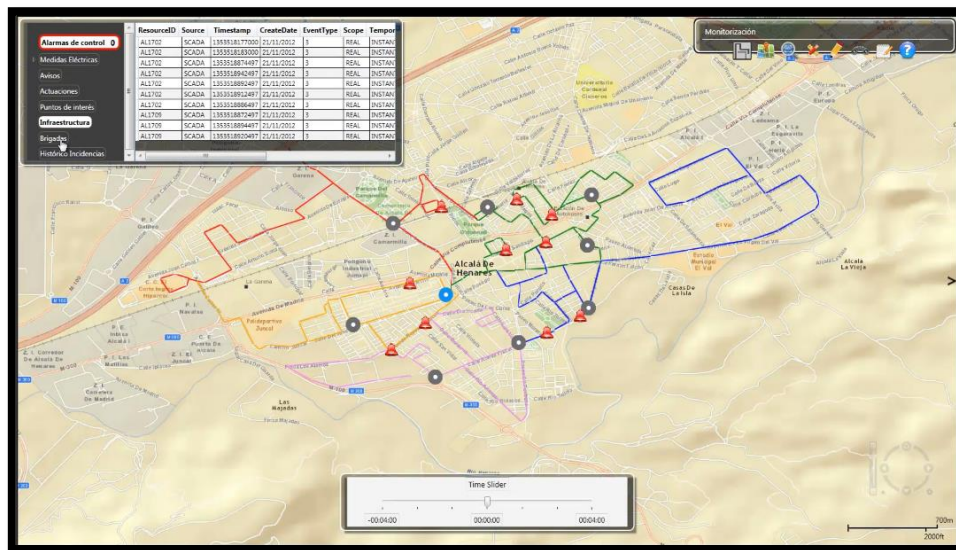


Figura 12: Coordinación vista tabular y visión espacial.

En la visión espacial los elementos más importantes son las alarmas y las estaciones. Por ello, además de la visión espacial y la visión tabular, se ha incluido otro apoyo más para la consulta de información de los elementos incluidos en estos grupos. En el caso de las alarmas, al seleccionar

una desde la visión espacial, aparecerá seleccionada en la visión tabular, se hará visible un *tooltip* que muestre los campos más importantes de la alarma.

El *tooltip* estará situado en la esquina superior de la alarma de manera que el operador pueda relacionar sin equivocación a qué alarma pertenece la información mostrada. El estilo de este componente se basa en el tema elegido para el resto de componentes. El fondo del *tooltip* será semitransparente para que el operador no pueda perder consciencia de lo que ocurre puesto que este componente se despliega sobre la visión espacial y esto puede llevar a que se interponga entre elementos de red y el operador.

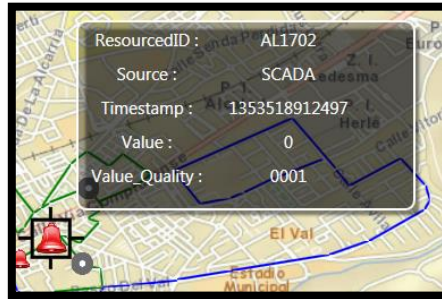


Figura 13: *Tooltip* alarma.

En el caso de los elementos de la infraestructura de red, estaciones y subestaciones, se añade un menú contextual que permita filtrar y conocer a mayor detalle los datos de un elemento en concreto. La manera de acceder a este menú contextual es la pulsación con el botón derecho del ratón sobre el elemento del que se desea conocer la información. El menú contextual que se despliega es un menú radial, como se muestra en la Figura 14. La elección de diseñar este tipo de menú viene condicionada por la ley de Fitts [14]. Esta ley establece el tiempo necesario que requiere el usuario para poder desplazarse desde un punto inicial hasta el punto final. Por lo tanto, al aplicar un menú radial se disminuye el tiempo que requiere el usuario para poder seleccionar una opción del menú, ya que todas las opciones se encuentran a la misma distancia.



Figura 14: Menú radial.

En el menú radial se encuentran tres opciones. Por cada opción se muestra un panel que lleva el identificador del elemento seleccionado puesto que los paneles se pueden mover por toda la

zona de monitorización, así el operador no puede dudar de a qué elemento corresponde la información mostrada. Las opciones del menú radial son las siguientes:

- **Notas:** Esta opción permite la consulta de las notas realizadas por todos los operadores en el elemento seleccionado. Al elegir esta opción se muestra un bloc de notas, el cual se muestra en la [Figura 15](#), basado en el tema base de toda la interfaz de usuario. En él se pueden consultar las notas según la clasificación de desde dónde se realizaron, por ejemplo desde la vista del mapa, desde la vista posicional, etc. Esta clasificación es importante ya que en cada vista se muestran unos datos diferentes por lo que el operador relaciona la nota escrita por otro o de él mismo con los datos que se encuentran en la vista de origen. Desde el componente para las notas se podrá crear o eliminar las notas que el operador considere oportunas. A este componente también se puede acceder, de manera general, desde el icono de notas del menú principal de la zona de monitorización.

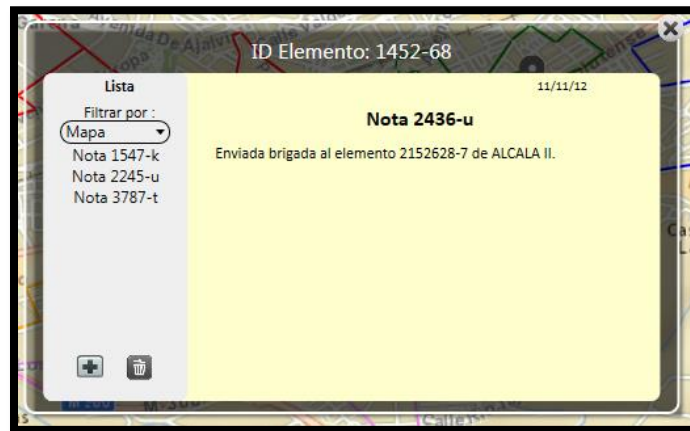
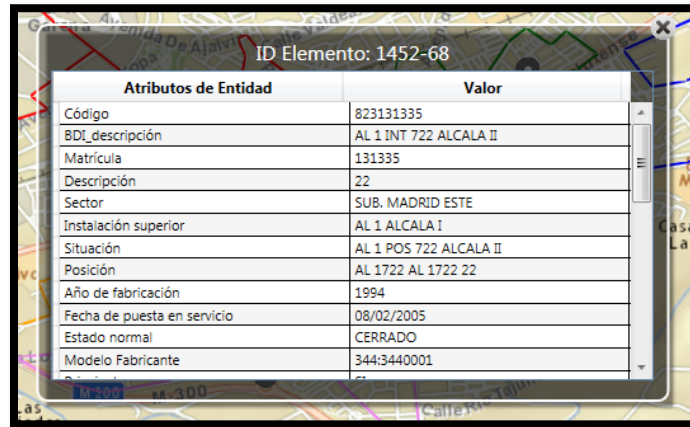


Figura 15: Bloc de notas de un elemento.

- **Detalle:** Este componente permite al operador conocer todos los datos específicos del elemento seleccionado. Los datos alfanuméricos que se muestran son los correspondientes a las características propias de cada elemento: estación o subestación. El panel desplegado está compuesto por un elemento *dataGrid*, como se aprecia en la [Figura 16](#), que ejerce de tabla para la colocación de los atributos y su valor.



Atributos de Entidad	Valor
Código	823131335
BDI_descripción	AL 1 INT 722 ALCALA II
Matricula	131335
Descripción	22
Sector	SUB. MADRID ESTE
Instalación superior	AL 1 ALCALA I
Situación	AL 1 POS 722 ALCALA II
Posición	AL 1722 AL 1722 22
Año de fabricación	1994
Fecha de puesta en servicio	08/02/2005
Estado normal	CERRADO
Modelo Fabricante	344:3440001

Figura 16: Componente detalle de un elemento.

- **Histórico:** Al seleccionar esta opción en el menú radial, se muestra un componente visual como el de la Figura 17. El objetivo de este componente es mostrar al operador de forma gráfica los valores eléctricos tomados en un rango de tiempo determinado. En el componente se establece un *tab panel* que permite la clasificación de los tipos de valores medidos, incluyendo una pestaña que recoja todos los valores para tener una visión global de las medidas.

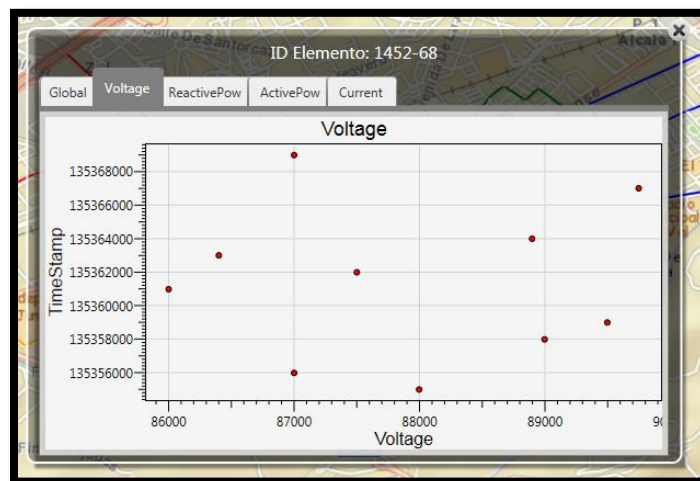


Figura 17: Gráfico de medidas eléctricas.

En la parte inferior de la zona de monitorización se encuentra un *slider* que define momentos en el tiempo. Este componente es la integración completa de la zona de proyección que fue disuelta en la primera iteración del ciclo de vida. Tras la realización del primer prototipo se disolvió esta zona ya que una de sus dos características principales, detalles específicos de determinados elementos, se cubría con la zona de control. La segunda característica, una proyección de lo que el sistema calcula que puede ocurrir según los datos recogidos, es la que se aborda con este componente, el *slider*. De esta forma, el componente se encuentra por defecto con el puntero

situado en el centro de la barra de tiempo, este punto es el equivalente al estado actual de la red y sobre el que se realizan los eventos que puedan surgir. En caso de mover el puntero hacia la izquierda, se mostrará en la zona de monitorización el mapa referente a lo ocurrido en el minuto atrás que señale el puntero. Dicho de otra forma, se mostrarán las alarmas o eventos que hayan ocurrido X minutos atrás. En caso de que el operador mueva el puntero hacia la derecha se mostrará una simulación (la proyección de eventos) de lo que podría ocurrir X minutos en adelante. Tanto para consultar lo ocurrido ya como para consultar la proyección existe un máximo de 4 minutos. Se ha determinado este número de minutos porque un operador siempre trata con eventos dentro de un rango de 5 minutos, a partir de ese tiempo el evento es considerado por el operador como un evento que lleva demasiado tiempo en el sistema. Para dar consciencia al operador de lo que está haciendo y del estado en el que se encuentra el sistema, siguiendo la primera heurística de Nielsen [15], cuando el operador seleccione ver la proyección de lo que calcule el sistema que puede ocurrir se ensombrece el mapa de monitorización y se muestra un letrero que indica que se trata de una simulación. Esta indicación se puede ver en la [Figura 18](#).

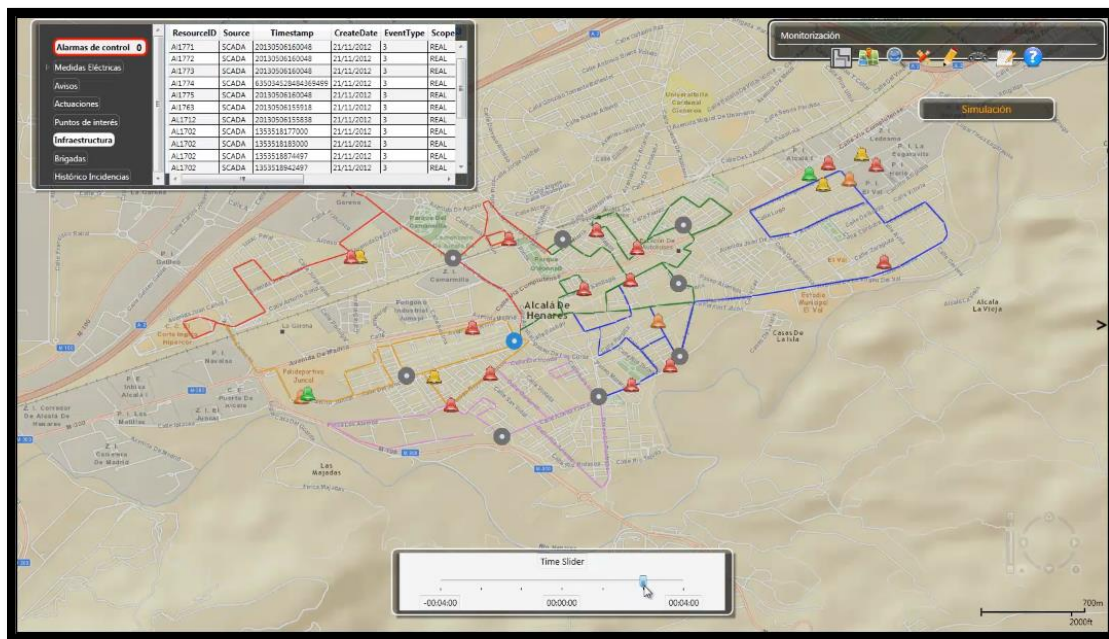


Figura 18: Visualización de simulación.

Las alarmas que se muestran en la zona de monitorización tienen dos tipos de visualización: por datos alfanuméricos o por situación geoespacial. Para la representación de este tipo de eventos se ha seguido un estándar, para dar consistencia siguiendo la cuarta heurística de Nielsen [15], para que el operador pueda identificar de forma ágil de lo que se le pretende informar. El estándar seguido es el estándar **Abnormal Situation Management** (ASM) [6]. Este estándar define como forma básica para la visualización de alarmas una campana. Para otorgar la prioridad a cada

una de las alarmas que se muestren en la monitorización de red, según el estándar ASM, se definen 4 colores:

Color	Prioridad
Rojo	Emergencia
Naranja	Alta
Amarillo	Media
Verde	Baja

Tabla 11: Relación color y prioridad de alarmas.

Una vez se tiene definido el tipo de representación que conllevan las alarmas, se diseña cómo se han de mostrar cuando se trata de alarmas nuevas. Esta parte tiene una gran importancia ya que es uno de los puntos clave para que el operador pueda adquirir el SA. Cuando el operador está interactuando con cualquier herramienta de la interfaz de usuario y surge una alarma nueva este evento debe resaltarse para que el operador pueda ser consciente de lo que ocurre. La forma de resaltar este tipo de eventos ha sido incluir un círculo negro que decolore hacia los bordes alrededor del icono de la campana de la alarma nueva, se aplica a todas las prioridades, como se muestra en la [Figura 19](#).



Figura 19: Sombreado alarma.

Cuando surge el evento el icono de la [Figura 19](#), parpadeará y en la visión tabular se indicará que existe un evento nuevo al lado de la etiqueta que marca la tabla de alarmas. También surgirá una nueva fila (en la tabla de alarmas) que contenga la información alfanumérica de la alarma nueva. Esta fila tendrá un color de fondo igual que el color que tenga la alarma en la visión espacial, es decir, el color de prioridad. Tras 10 segundos y, a la par que cuando el icono deja de parpadear, el color de fondo de la fila de la alarma nueva perderá intensidad, mostrando así el mismo color pero más pálido. De esta forma, el operador sabrá reconocer que se trata de una alarma nueva pero que no acaba de surgir sino que han pasado 10 segundos ya desde que se declaró. De la misma manera, y tras haber ocurrido esto, al pasar de nuevo 10 segundos (20 segundos desde que se inició la alarma) en la visión tabular la fila correspondiente perdería todo el color y en la visión espacial el icono de la alarma continúa con el sombreado negro. Cuando surge otra alarma nueva el icono de la anterior perderá el sombreado quedando únicamente la campana como tal. Esta secuencia de acciones se resume en la [Tabla 12](#):

Visión alarma	Inicio alarma	Tras 10 segundos	Tras 20 segundos	Otra alarma nueva
Visión espacial	Icono sombreado y parpadeo.	Icono sombreado.	Icono sombreado.	Icono normal.
Visión tabular	Se indica el número de alarmas nuevas y se añade la fila de datos con color de fondo igual que el color de la campana.	Se resta la alarma al número de alarmas nuevas y el color de fondo de la fila pierde intensidad.	La fila de datos de la alarma pierde todo el color y queda blanca como el resto.	Fila de datos con fondo blanco.

Tabla 12: Secuencia de acciones con alarma nueva.

A continuación, en la [Figura 20](#), se muestra cómo pierden color las filas de la visión tabular. En la imagen superior se muestra la visión tabular con el color intenso cuando las alarmas son nuevas mientras que en la imagen inferior se muestra la misma visión tabular tras el paso de los primeros 10 segundos. También se puede apreciar cómo varía, según el tiempo desde que surgieron las alarmas, el contador que se encuentra en la etiqueta de la tabla de datos de “Alarmas de control”.



Figura 20: Contraste de color en visión tabular.

Otra herramienta que se ha incluido en la zona de monitorización es la selección de una única línea (“selección de línea”). Esta herramienta parte de la selección de área con la diferencia de que en este caso en el área dibujada se aplicará un filtro para obtener la línea que mayor espacio ocupa en el área. A la hora de seleccionar un área o una línea se diferencian los rectángulos dibujados mediante el color. En el caso de la selección de área el rectángulo es gris mientras que en el caso de la selección de línea el rectángulo es azul. Ambas selecciones también se diferencian por la información que se muestra en la zona de control. Además del color de la selección la principal diferencia reside en la información mostrada para cada una en la zona de control, como se puede observar en la [Figura 21](#).

Para la selección de línea se muestran los elementos necesarios para tener un control mayor de un línea concreta. El principal elemento es un mapa ortogonal de la línea filtrada, con ello se podrá observar los diferentes enlaces y terminaciones que engloba la línea. También se muestra un árbol de componentes dentro de la infraestructura, de tal forma que el operador pueda conocer la jerarquía de componentes que tiene la línea. Por último, se filtra del histórico de incidencias un listado que incluya todas las incidencias ocurridas en la línea en un rango de tiempo determinado.

Para la selección de área se muestran elementos que permiten tener una monitorización específica de una zona. Entre ellos se muestra el área que ha sido seleccionada, recortada del mapa global. Sobre esta área se pueden aplicar también los filtros de capas que existen en la zona de monitorización, el operador podrá activar o desactivar capas mediante el menú principal de la zona de control. Se incluye también una ventana que aporte una imagen del mapa ortogonal de las líneas incluidas en el área seleccionada. A la par que ello, se filtran las incidencias correspondientes a estas líneas y se muestra una tabla que las recoja.



Figura 21: Selección de línea y selección de área.

En la zona de control se incluye la opción de limpiar el contenedor de las ventanas, de manera que el operador pueda dejar la zona de control libre y poder expandir nuevos datos de elementos o áreas diferentes. Como ya se ha dicho en el párrafo anterior, cuando se selecciona un área en la zona de monitorización, se crea una ventana en la zona de control que contiene el recorte del área seleccionado y que continúa coordinado con el mapa de monitorización global. Esta coordinación permite al operador tratar la zona específica con datos actualizados. El objetivo de la selección de un área es filtrar la información a una zona específica por ello en la zona de control se le da la opción al operador de poder interactuar con el mapa recortado. El operador puede aplicar zoom o movimiento por el mapa recortado pero también puede aplicar un filtro de capas de elementos de red. Desde el menú principal de la zona de control, como se muestra en la [Figura 22](#), el operador puede hacer visibles u ocultar las capas que necesite para poder tratar la información a un mayor nivel de profundidad de la zona seleccionada.

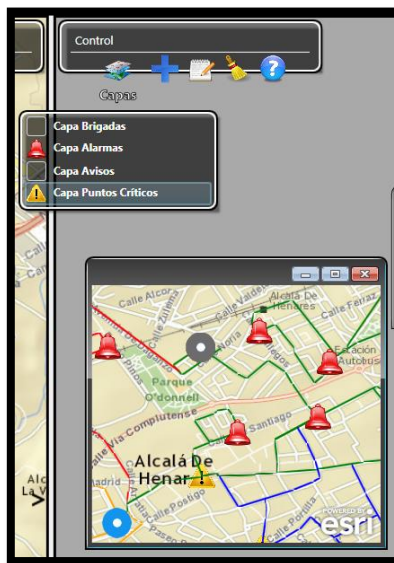


Figura 22: Selección de capas en zona de control.

En el menú principal de la zona de control se ha incluido un icono que representa el símbolo “+”. Este icono sirve para la creación de nuevas actuaciones a partir de los datos que se encuentran contenidos en la zona de control. Tal y como especifican los requisitos de usuario, al crearse una actuación nueva ésta será automáticamente del tipo “No programada”, aplicándole el color relacionado a este tipo, el color rojo. Por lo tanto, como se muestra en la [Figura 23](#), al crear una nueva actuación se cambia el color del contenedor de la zona de control, se le otorga a la actuación un nuevo identificador y se incluye el objeto dentro de la clasificación de “No programadas” en la zona de actuaciones.

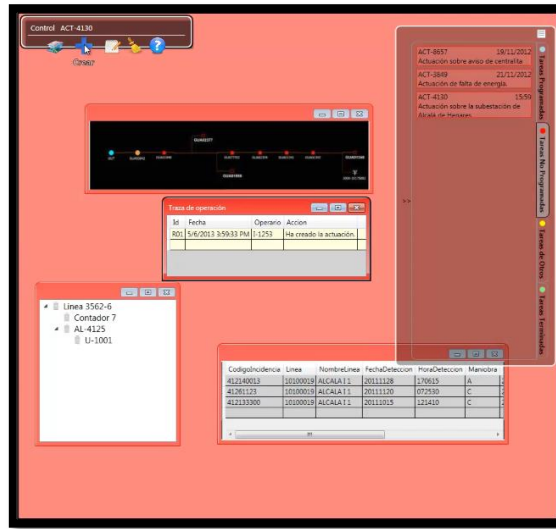


Figura 23: Crear actuación.

Al crear el operador una nueva actuación se inicializa un registro o *log* que tiene por objetivo recoger todas las acciones o comentarios acerca de la actuación. Por defecto se inicializa este *log* con un primer registro que indique la fecha y hora de creación de la actuación junto al identificador del operador que la ha generado. A partir de ahí, el operador podrá modificar cada entrada que se incluya en el *log* añadiendo los datos que considere oportunos e importantes para el desarrollo de esa tarea. Para poder incluir nuevas acciones, el operador únicamente deberá seleccionar la fila que se encuentra vacía y la interfaz cambiará señalando al operador que ya puede introducir texto y en que columna se va a incluir. El *log*, el cual se muestra en la Figura 24, servirá de control de acciones dentro de una tarea con el fin de facilitar la comunicación indirecta entre operadores que intervengan en esa actuación. El formato del *log* sigue el estilo de una tabla simple. Para poder diferenciarlo de otras tablas, que se puedan incluir en la actuación (por ejemplo la tabla de filtrado de incidencias), se incluye en una ventana denominada “Traza de operación” y el color de fondo de las filas será un amarillo pálido a fin de simular las “anotaciones” que se puedan realizar en la vida real.

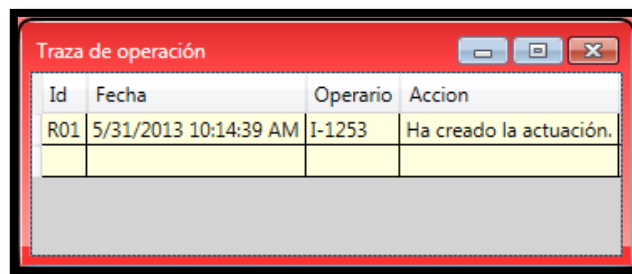


Figura 24: Log de operaciones.

En la zona de comunicación y de actuaciones no se ha incluido nada nuevo según el prototipo de la tercera fase.

Con el diseño de la interfaz de usuario obtenido en esta última iteración en el ciclo de vida del producto se cubren todos los requisitos propuestos en la especificación de requisitos.

Implementación

El prototipo software se ha llevado a cabo en mediante la tecnología *Windows Presentation Foundation*, utilizando como entorno de programación el software Visual Studio 2012.

La tecnología WPF presenta diferentes ventajas frente al resto de tecnologías comparadas pero las dos principales características a tener en cuenta son: el patrón de diseño **Model-View-ViewModel** (MVVM) y la capacidad de diseño en los componentes visuales. En cuanto al plano de la implementación el patrón de diseño que se aplica es la característica más importante ya que influye tanto en la ejecución de la interfaz como en la organización del proyecto.

El modelo arquitectónico MVVM propone una estructuración con la ventaja de que se puede diferenciar entre la parte visual, funcional y de acceso a datos, de cada subsistema sin que haya una gran dependencia entre cada una de ellas. Este modelo arquitectónico ofrece un nuevo módulo con respecto al clásico Modelo-Vista-Controlador, el **ViewModel**.

El módulo **ViewModel** contiene todo el negocio de la vista con la que se relaciona. Esto quiere decir que, éste módulo contiene toda la funcionalidad de la vista relacionada con los modelos de datos y las operaciones aplicadas a ellos. Las únicas funcionalidades que no se incluyen en este módulo son las que realizan cambios en los componentes visuales de la vista.

El módulo **View**, corresponde con la parte visual. Contiene todos los componentes gráficos. Éste módulo está dividido en dos partes. Por un lado, está la vista propiamente dicha que contendrá los estilos y los componentes visuales. Por otro lado, está una clase asociada a cada una de estas vistas que contengan todas las operaciones que modifican la visualización mostrada por los componentes visuales.

El módulo **Model**, contiene todas las clases necesarias para el acceso y la conversión de datos a los formatos necesarios para su tratamiento. En este sistema, este módulo realiza una conexión y consulta a una base de datos desde un servidor en red. También accede a diferentes ficheros Excel para realizar una conversión de datos al formato *dataSet*, propio de WPF.

A continuación, en la [Figura 25](#), se muestra un esquema de cómo está implementado este modelo arquitectónico. Indicando el flujo de datos entre los diferentes módulos.

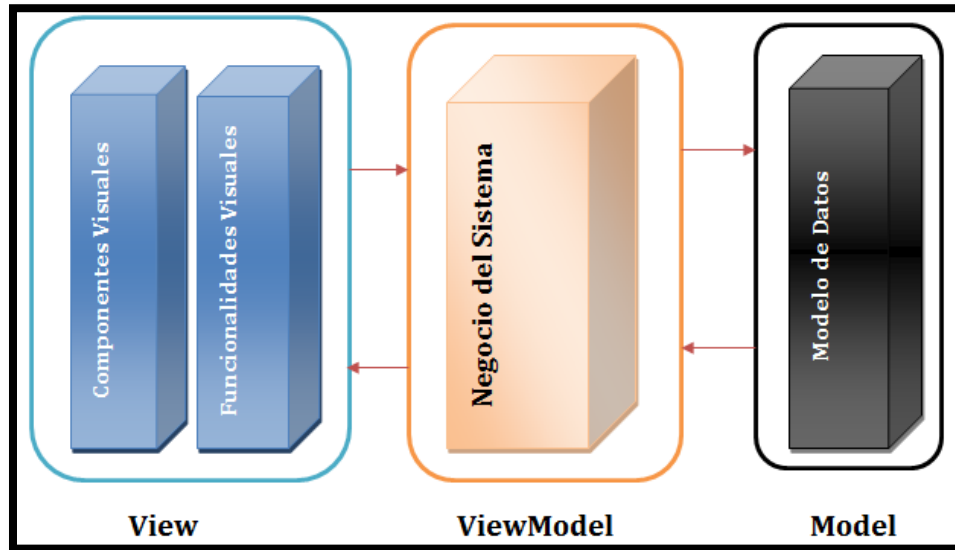


Figura 25: Patrón Model-View-ViewModel.

En la Figura 26, se muestra un diagrama estructural de cómo se han organizado los componentes de la interfaz de usuario. En el diagrama se aprecia que se tratan nombres conceptuales y no técnicos, ya que el objetivo de este esquema es mostrar la filosofía seguida para la organización según el patrón MVVM.

En el comienzo del diagrama, Figura 26, se muestran las zonas en las que se encuentra dividida la interfaz y de ellas parten las tres vistas principales de la aplicación. En la figura se identifican estos componentes por el color azul oscuro. A estas vistas principales se les asocia un **ViewModel** que contenga toda la funcionalidad y el negocio de la zona de la interfaz de la que se trate. Las **View Overview** y **View Detail** no albergan directamente componentes visuales como tales sino que se componen de otras vistas. En estos dos casos, la vista general sirve para agrupar de manera jerárquica las vistas incluidas en una misma zona. La vista general también tiene la asociación al **ViewModel**, por lo tanto al incluir otras vistas, éstas pueden acceder a la información y funcionalidad del **ViewModel** al que está asociada la vista general de la zona. Estas vistas situadas un nivel más abajo en la organización se identifican en el esquema por el color azul claro. Esta arquitectura, junto con las posibilidades que ofrece la tecnología, también presenta nuevos elementos llamados **UserControls**. De esta forma, se designa a los elementos que contienen componentes creados por el desarrollador, es decir, cada vez que el diseñador genera un nuevo componente visual, siguiendo el patrón arquitectónico, este será independiente del resto de componentes y será de tipo **UserControl**. En el diagrama de la Figura 26 se muestran los conceptos que son tratados como un **UserControl** en la implementación de la interfaz de usuario, estos componentes se identifican con el color verde claro.

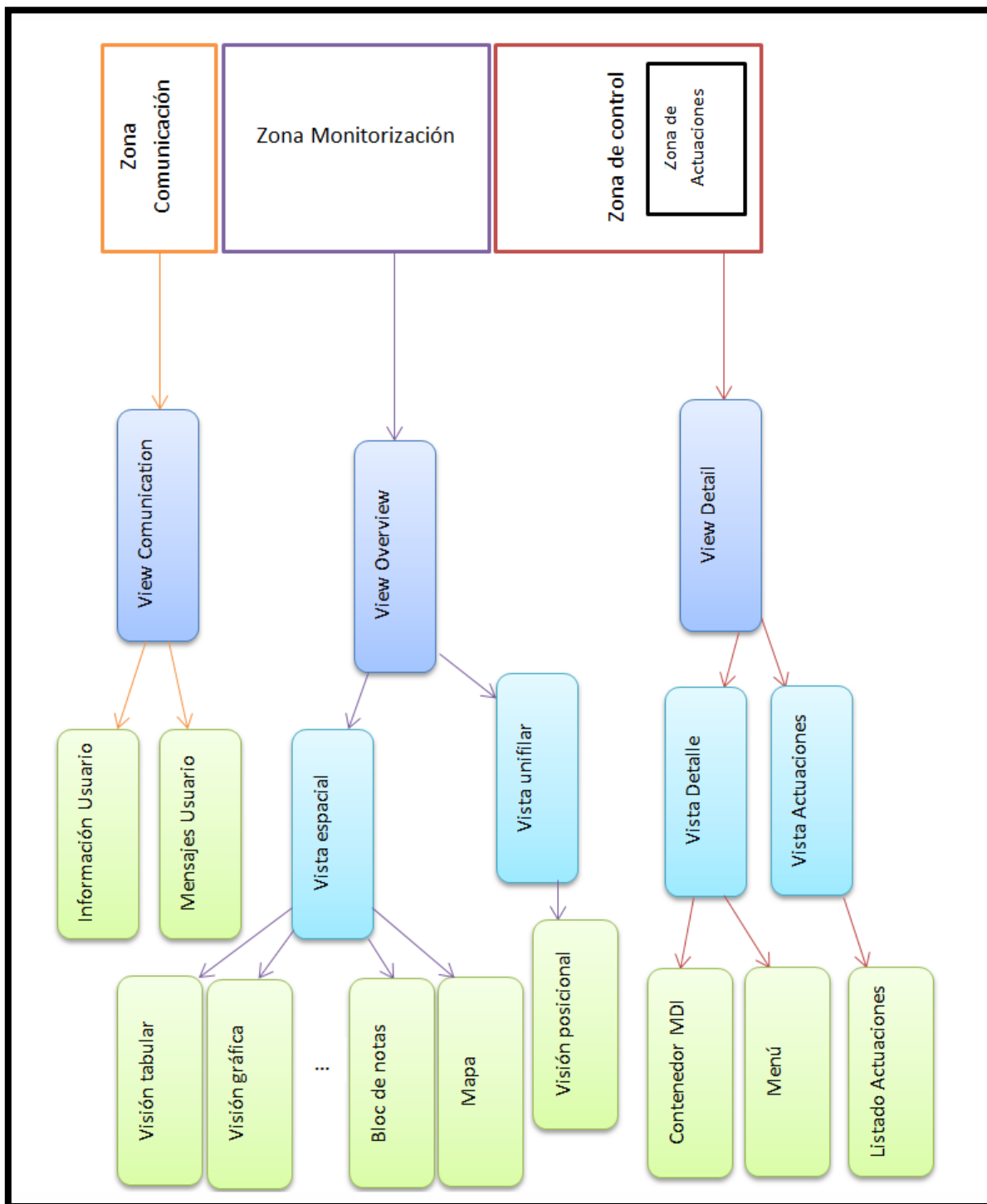


Figura 26: Organización MVVM.

Siguiendo la estructura de MVVM se ha organizado el proyecto con una estructura de carpetas que respete la independencia de los diferentes módulos necesarios.

La solución ENERGOS está formada por la conjunción de tres proyectos. Dos de estos tres proyectos son complementarios al tercero. Esto quiere decir que los proyectos de GongSolutions.Wpf.DragDrop y WPF.MDI son proyectos incluidos para poder realizar o implementar una funcionalidad en el proyecto principal, Energoss.

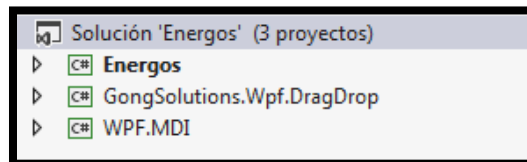


Figura 27: Solución Energoss.

El proyecto GongSolutions.Wpf.DragDrop está compuesto por una serie de clases que proporcionan los útiles necesarios para poder implementar la herramienta de *DragAndDrop* en el proyecto principal. Por otro lado, el proyecto WPF.MDI implementa las condiciones necesarias para la creación de las ventanas independientes propias de Microsoft. De esta forma, en el proyecto principal se pueden declarar cada tipo de ventana según tipo de contenido que deba contener: tabla, imagen, árbol, etc.

En la Figura 28, se puede observar el contenido completo de cada uno de estos proyectos.

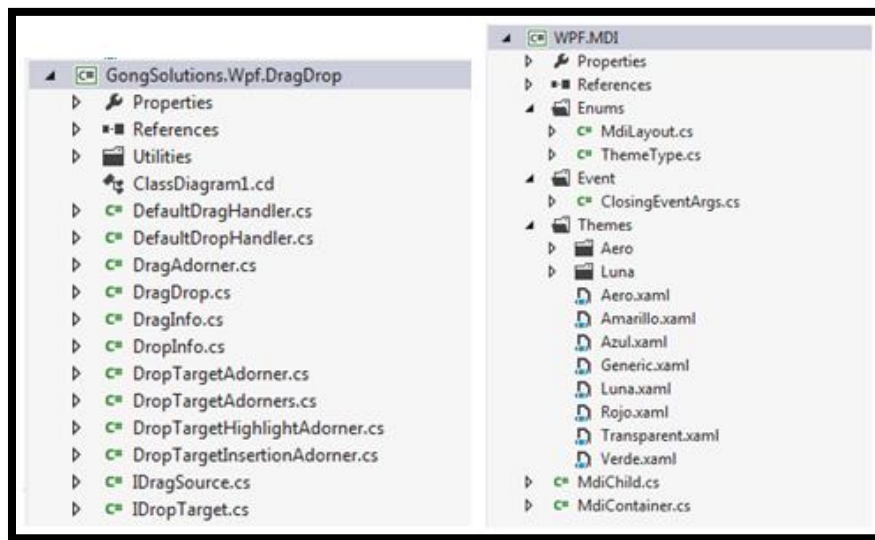


Figura 28: Estructura de proyectos auxiliares.

El proyecto principal, Energoss, se compone de una serie de carpetas que indican la utilidad de las clases que contienen.

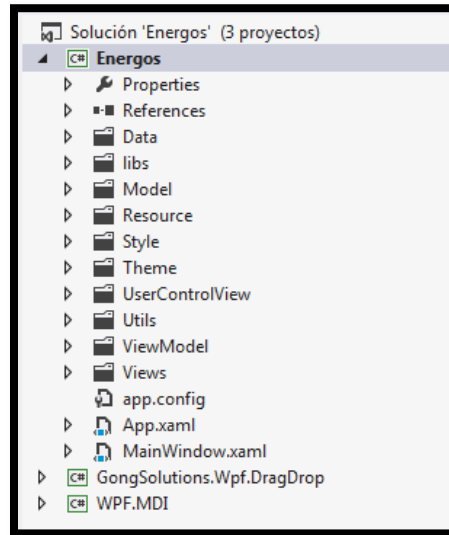


Figura 29: Estructura Energos.

Entre los componentes del proyecto destacan las carpetas “Views”, “ViewModel”, “UserControlView” y “Model”. Estas son las carpetas que contienen los componentes definidos en el diagrama de la Figura 26 siguiendo el patrón MVVM.

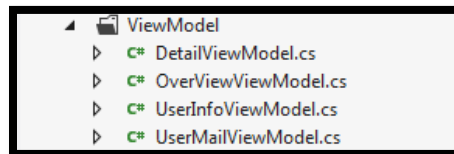


Figura 30: Carpeta ViewModel.

En el caso de la carpeta “UserControlView” se establece una estructura de carpetas según las zonas de la interfaz de usuario. Los componentes de esta carpeta son una gran cantidad debido a que son todos los componentes visuales que contiene la interfaz de usuario. Se establecen como subcarpetas “Action”, “Comunication” y “OverView” para los componentes específicos de cada zona. También se establecen las carpetas “EnergosControl” y “Theme” para componentes que funcionan de manera genérica para las cuatro zonas.

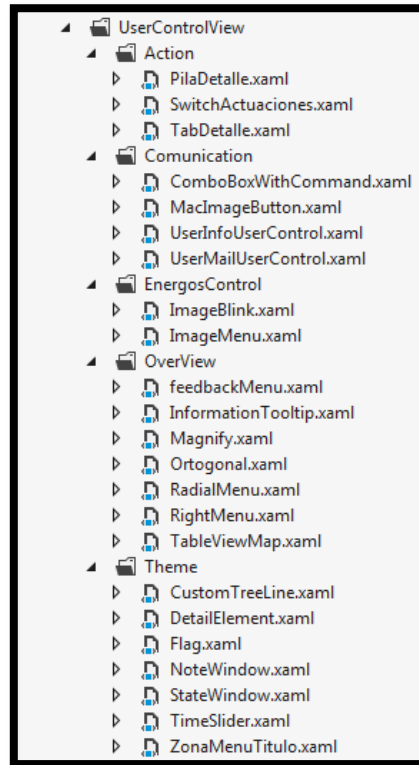


Figura 31: Carpeta UserControlView.

La carpeta de “Views”, como se muestra en la [Figura 32](#), sigue una estructuración con subcarpetas que limitan las zonas que son definidas en la interfaz. En este caso se hace referencia a la zona de actuación con el nombre de “Acción”.

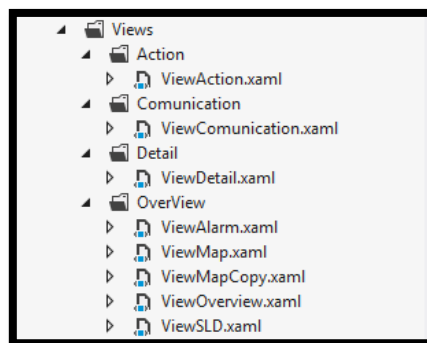


Figura 32: Carpeta Views.

La carpeta “Model” está estructurada según los tipos de datos que necesita la interfaz de usuario. Estos datos se diferencian entre objetos de origen MySQL o Excel. Por cada tipo de objeto se crea un tipo de *dataSet* que acoja todas las características del objeto en su lugar de origen. Para

transformar este tipo de dato en el necesario por la interfaz de usuario, esta clase se denomina con el nombre del objeto seguido de las iniciales de *Business Object* (BO).

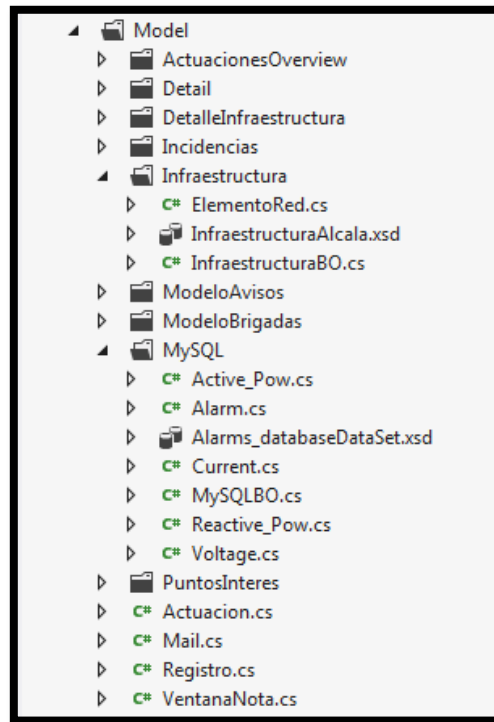


Figura 33: Carpeta Model.

Además de las clases principales, en la estructura del proyecto Energoss, se encuentra la definición de las referencias que tiene el proyecto principal a diferentes archivos DLL que permiten desde la conexión a la base datos como la visualización del mapa. Este apartado resulta ser de los más importantes dentro del proyecto porque es el que define las restricciones de integración de la interfaz de usuario. Esto es debido a que el proyecto depende de las referencias que se encuentren definidas y a las que pueda acceder independientemente del ordenador en el que se esté ejecutando. Es en este mismo apartado donde se indica la referencia a los dos proyectos auxiliares para que el principal pueda ejecutar componentes o utilidades de ellos.

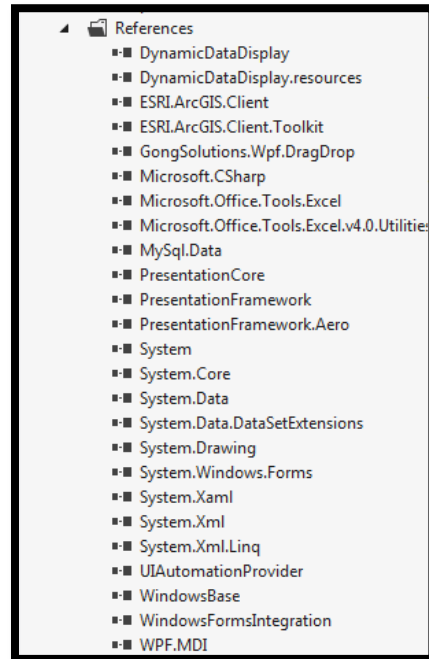


Figura 34: Carpeta de referencias.

Por último, en la carpeta de “Utils” se incluyen todas las clases que son una utilidad de manera genérica para todo el proyecto, y que no tienen ningún tipo de conexión con el aspecto visual de la interfaz sino que son las que apoyan a la funcionalidad.

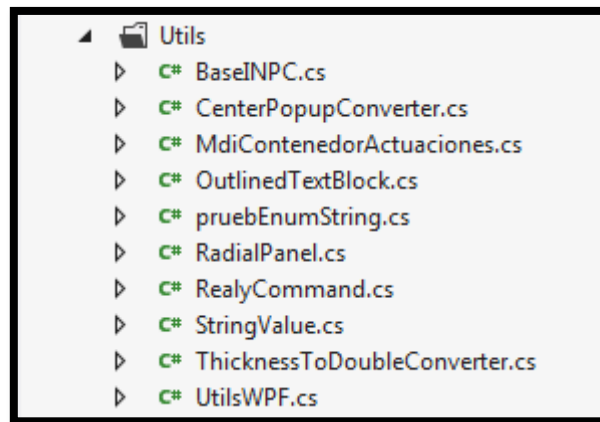


Figura 35: Carpeta Utils.

5 Evaluación

En esta sección se exponen los métodos llevados a cabo para la evaluación de la solución y los resultados aportados. El objetivo de realizar una evaluación es demostrar la validez de la solución elaborada. Para ello se realiza un proceso de evaluación mediante diferentes pruebas y se considera que la solución es válida si se comprueba que presta solución a los problemas expuestos en el planteamiento del problema y satisface los objetivos definidos para el mismo.

A lo largo de esta sección se va a definir el formato de pruebas a realizar para validar la solución propuesta y se especificará los casos de pruebas concretos que se van a llevar a cabo por los usuarios. Tras la exposición de todos los casos de prueba se mostrará una matriz de trazabilidad entre los requisitos de la solución y los casos de prueba. Por último, se realizará un análisis de los resultados recopilados tras la realización de dichas pruebas.

5.1 Proceso de evaluación

Plan de pruebas

La solución propuesta es un prototipo software de la interfaz de usuario de un sistema para el control de la red eléctrica. El objetivo de este diseño es apoyar al operador para adquirir SA a la hora de monitorizar la red eléctrica. Para poder evaluar la solución se plantean cuatro escenarios que los usuarios deben seguir, interactuando con la interfaz de usuario, para alcanzar uno o varios objetivos.

La especificación de los escenarios se realiza utilizando la plantilla que se muestra en la [Tabla 13](#).

Nombre del escenario	
Identificador de escenario	
Descripción del escenario	
Tareas a realizar	
Verificación de tareas	

Tabla 13: Plantilla para escenarios.

Por cada escenario se indica el nombre del mismo y se le otorga un identificador. El identificador tendrá el siguiente formato “ESC-XXX”, dónde la parte de las equis hacen referencia a la parte numérica del identificador. En cada escenario se definen tres partes importantes, en primer lugar se plantea una parte narrativa que exponga el caso del escenario y una breve descripción. En segundo lugar, se definen una serie de pasos más concretos acerca de las tareas a realizar por el usuario hasta llegar al fin deseado. El cometido de esta segunda parte es desgranar la narrativa expuesta en la primera. En tercer lugar, se plantea una serie de acciones o visualizaciones que deberán ocurrir cuando el usuario realice las tareas propias del escenario. Esta parte es la correspondiente a la verificación de las pruebas, así teniendo según las tareas clave que ha de realizar el usuario se dictan lo que serán las consecuencias de las mismas. De esta forma, si

al realizar la prueba no se obtienen las acciones definidas en el apartado de verificación no se podrá contar como una prueba sin errores.

Casos de prueba

Para evaluar todo el conjunto de requisitos funcionales se diseñan cuatro escenarios que alberguen una serie de tareas que al cumplir el usuario permitan la verificación de los requisitos. A continuación se especifican cada uno de estos escenarios y, al final de esta sección, se muestra la matriz de trazabilidad entre los requisitos funcionales de la solución y los escenarios descritos.

Nombre del escenario	Selección de área
Identificador de escenario	ESC-001
Descripción del escenario	El operador en estado de monitorización de la red estará pendiente de los avisos que puedan aportar los clientes de la red eléctrica. En un momento determinado aparece en la red una alarma nueva con prioridad media. El operador, al no tratarse de una alarma con prioridad baja, consulta la proyección del sistema sobre las posibles alarmas que pudieran surgir en los próximos minutos para razonar si puede tratarse de un hecho aislado o el inicio de una cadena de alarmas. Ante la posibilidad de esta segunda opción, el operador decide adelantarse a los hechos y selecciona el área que se vería afectada en los siguientes minutos. Una vez ha filtrado la información acerca de esa área la examina con máximo detalle con la intención de encontrar el estado de los elementos. Observa que puede existir un elemento con el peligro de crear un fallo en cadena y decide crear una nueva actuación con toda la información recopilada para que sus compañeros tengan constancia de la situación.
Tareas a realizar	Listado de tareas: <ul style="list-style-type: none"> • Observar la aparición de una nueva alarma. • Consultar el histórico visual de alarmas y la simulación para los siguientes minutos. • Seleccionar un área. • Comprobar en la zona de control los diferentes datos del área seleccionado: medidas eléctricas, posicional, elementos incluidos en esa zona, etc. • Crear una actuación con la información recopilada.
Verificación de tareas	Listado de acciones que verifican las tareas: <ul style="list-style-type: none"> • Se muestra el icono de una campana parpadeando en la zona espacial y los datos alfanuméricos de la alarma nueva. • Al desplazar el slider del tiempo en la visión espacial se muestran las alarmas anteriores al momento actual. Al

	<p>mover el slider hacia delante, se ensombrece la visión espacial y se muestran las posibles alarmas que surjan.</p> <ul style="list-style-type: none"> • Al seleccionar un área se muestra un rectángulo que ofrece un feedback al operador para que conozca el área que está seleccionando. • Al seleccionar un área, en la zona de control se muestran varias ventanas que contienen: el mapa del área, listado de incidencias y la visión posicional. • Al pulsar sobre el icono del “+” del menú principal de la zona de control, ésta cambia el color de fondo a color rojo y se añade una ventana llamada “Traza de operación” con el log de operaciones de la actuación creada. Esta actuación también se añade en el listado de actuaciones de tipo no programadas.
--	--

Nombre del escenario	Monitorización de la red
Identificador de escenario	ESC-002
Descripción del escenario	<p>El operador se encuentra monitorizando la red eléctrica mediante la observación del mapa posicional de la misma. Y decide cambiar a la visión espacial para poder comprobar de manera geoespacial la situación de las brigadas y los avisos nuevos de los clientes. Al observar que un aviso se encuentra cerca de una subestación, el operador consulta la información en detalle de esta infraestructura. Los datos que observa son normales por lo que decide cambiar la visión espacial a tipo de relieve. De esta forma, podrá comprobar si algún factor medioambiental ha podido ser la causa del aviso del cliente. El operador, en este caso, no encuentra ningún tipo de relación evidente por lo que decide consultar las medidas eléctricas tomadas en la subestación en un rango de varios minutos para comprobar si existe algún pico de tensión o voltaje. Como no observa nada fuera de lo normal, decide consultar las notas del elemento para ver si otro compañero lo tiene bajo vigilancia. Al comprobar que un compañero ya tiene el elemento vigilado, el operador continúa con su tarea inicial de monitorización.</p>
Tareas a realizar	<p>Listado de tareas:</p> <ul style="list-style-type: none"> • Visualizar la red con la visión posicional de la misma. • Cambiar la visualización a la visión espacial. • Activar/desactivar capas de los elementos de red.

	<ul style="list-style-type: none"> • Seleccionar un elemento de la infraestructura de la red. • Consultar el menú contextual del elemento seleccionado. • Consultar los detalles del elemento. • Cambiar la visión espacial a tipo relieve. • Consultar los gráficos de medidas eléctricas del elemento seleccionado. • Acceso a notas del elemento.
Verificación de tareas	<p>Listado de acciones que verifican las tareas:</p> <ul style="list-style-type: none"> • Al pulsar sobre el icono de “Posicional” del menú principal de la zona de monitorización se muestra la visión posicional de la red eléctrica. • Al pulsar sobre el icono de “Posicional” del menú principal de la zona de monitorización se oculta la visión posicional y se muestra la visión espacial de la red. • Al pulsar sobre cada uno de las etiquetas del árbol que se encuentra en la visión tabular de la zona de monitorización, se hace visible la capa que se haya seleccionado en la visión espacial. • Al pulsar sobre uno de los elementos de la infraestructura de red, éste aparece con un borde negro alrededor y se marca en la visión espacial los datos del elemento que ha sido seleccionado. • Al pulsar con el botón derecho del ratón sobre un elemento de la infraestructura de red de la visión espacial, se muestra un menú radial con las opciones “Notas”, “Detalle” e “Histórico”. • Al seleccionar del menú contextual la opción de “Detalle” se muestra una nueva tabla con el identificador del elemento y el valor de cada uno de los atributos del mismo. • Al pulsar sobre el icono de “Mapas” del menú principal de la zona de monitorización se despliega un submenú con las opciones “Callejero” y “Relieve”. Al pulsar sobre la segunda opción, la visión espacial cambia a una visión espacial con relieve. • Al seleccionar la opción de “Histórico” del menú contextual de un elemento de red, se muestra un componente que permite visualizar gráficos de los datos de medidas eléctricas recogidos. • Al seleccionar la opción de “Notas” del menú contextual de un elemento de red, se muestra un bloc de notas que

	permite visualizar las notas realizadas por el operador u otros operadores sobre el elementos seleccionado.
--	---

Nombre del escenario	Selección de línea
Identificador de escenario	ESC-003
Descripción del escenario	El operador se encuentra respondiendo un correo de otro compañero. Al finalizar el operador vuelve sobre la tarea de monitorización y para observar con mayor detalle los elementos de la red decide utilizar la herramienta de la lupa. Mientras el operador se encuentra realizando una monitorización exhaustiva de los elementos de red, llega un mensaje nuevo en la zona de comunicación. El operador lo visualiza y lee la respuesta de su compañero. En ese momento surge una nueva alarma en la visión espacial. El operador decide consultar los datos que se muestran en la visión tabular de esta nueva alarma, comprobando el tipo de alarma y los elementos sobre los que incide. Al ver que perjudica a una línea central de Alcalá de Henares, el operador selecciona la línea para poder examinar las incidencias ocurridas y los componentes que contiene esta línea, en la zona de control.
Tareas a realizar	Listado de tareas: <ul style="list-style-type: none"> • Responder mensaje. • Utilizar la herramienta de la lupa. • Leer mensaje. • Observar que aparece una nueva alarma en la visión espacial. • Consultar datos en la visión tabular. • Selección de línea. • Visionar datos específicos en la zona de control.
Verificación de tareas	Listado de acciones que verifican las tareas: <ul style="list-style-type: none"> • Al pulsar sobre el icono de responder, se despliega un panel translúcido sobre la zona de monitorización. En este panel, el operador introduce los datos que desee y pulsa sobre el botón responder. Al responder, el componente desaparece de la zona de monitorización. • Al pulsar sobre el icono de “Lupa” del menú principal de la zona de monitorización, se muestra una lupa de gran tamaño sobre la zona de monitorización. • Al aparecer un nuevo mensaje, se muestra el nuevo

	<p>mensaje en la bandeja de entrada con un color de fondo más oscuro que el que color de los mensajes ya leídos.</p> <ul style="list-style-type: none"> • Al aparecer una nueva alarma en la visión espacial, se muestra un icono de alarma sombreado que parpadea sobre la zona de monitorización. • Al aparecer una alarma nueva, se crea una nueva fila de datos en la visión tabular de las alarmas. Esta fila se colorea del color que indica la prioridad de la alarma. Al seleccionar el operador sobre esta fila, el icono de la alarma pasa a tener una mirilla negra sobre él. En la visión tabular, al estar seleccionada la alarma, se muestra la fila de datos con un color grisáceo. • Al pulsar sobre el icono de “Selección de línea” del menú principal de la zona de monitorización, se muestra al operador un feedback del rectángulo que dibuja para filtrar la línea más importante de la zona. • Al seleccionar una línea, en la zona de monitorización se muestran tres ventanas que contienen el árbol de componentes de la línea, la visión posicional de la línea y el listado de incidencias de la misma.
--	--

Nombre del escenario	Seguimiento de actuaciones
Identificador de escenario	ESC-004
Descripción del escenario	<p>El operador se encuentra monitorizando la red y al ver una alarma de prioridad alta decide consultar los datos de una alarma desde la visión espacial, la selecciona y valora los datos mostrados en el <i>tooltip</i> de la misma. Al comprobar que se trata de una alarma importante, el operador abre cada una de las actuaciones programadas para constatar que esa alarma está siendo seguida o tratada. Al no encontrar referencia a esta alarma en las actuaciones programadas, el operador decide consultar en las actuaciones no programadas por si se tratara de una alarma conocida recientemente. En este caso encuentra una actuación que hace referencia al seguimiento de dicha alarma y el operador añade un nuevo registro en el <i>log</i> de la actuación indicando, junto a su identificador, que él se encuentra vigilando las medidas eléctricas de dicho evento. Tras haber inscrito sus operaciones en <i>log</i>, limpia la zona de control y continúa con la monitorización de la red.</p>
Tareas a realizar	<p>Listado de tareas:</p> <ul style="list-style-type: none"> • Consulta del <i>tooltip</i> de una alarma.

	<ul style="list-style-type: none"> • Consulta de actuaciones programadas. • Consulta de actuaciones no programadas. • Edición de registro en el <i>log</i> de una actuación. • Limpiar zona de control.
Verificación de tareas	<p>Listado de acciones que verifican las tareas:</p> <ul style="list-style-type: none"> • Al seleccionar una alarma en la visión espacial de la zona de monitorización, se muestra un pequeño <i>tooltip</i> que contiene los datos específicos de la alarma seleccionada. • Al seleccionar la clasificación de “Programadas” en el listado de actuaciones, se muestra una pila de actuaciones de color azul que se corresponden con las actuaciones programadas de la zona. • Al arrastrar, el operador, una actuación sobre la zona de control se muestra en esta zona el contenido completo de esta actuación. También cambia el color de fondo de la zona de control a azul. • Al seleccionar la clasificación de “No Programadas” en el listado de actuaciones, se muestra una pila de actuaciones de color rojo que se corresponden con las actuaciones no programadas de la zona. • Al arrastrar, el operador, una actuación no programada sobre la zona de control se muestra en esta zona el contenido completo de esta actuación. También cambia el color de fondo de la zona de control a rojo. • En una actuación no programada, al pulsar en una fila del <i>log</i> contenido en la ventana de “Traza de operaciones”, ésta se sombrea de color azul y permite al operador introducir el texto para la edición del <i>log</i>. • Al pulsar sobre el icono de “Limpiar” del menú principal de la zona de control, se eliminan visualmente todo el contenido de la zona. Los datos siguen guardados, simplemente se despeja la zona y se devuelve a su estado por defecto.

A continuación, en la [Tabla 14](#), se muestra la matriz de trazabilidad entre los requisitos funcionales y los escenarios de prueba expuestos.

Requisitos/ Escenarios	ESC-001	ESC-002	ESC-003	ESC-004
FU-001	X	X	X	X
FU-002		X		
FU-003		X		
FU-004			X	
FU-005				X
FU-006				X
FU-007		X		
FU-008			X	
FU-009	X	X	X	
FU-010	X	X	X	
FU-011	X	X	X	
FU-012	X	X	X	X
FU-013			X	
FU-014			X	
FU-015			X	
FU-016			X	
FU-017	X			
FU-018			X	
FU-019	X			
FU-020	X			
FU-021	X		X	
FU-022	X		X	
FU-023	X		X	
FU-024	X		X	
FU-025		X		
FU-026		X		
FU-027		X		
FU-028		X		
FU-029		X		
FU-030	X	X	X	X
FU-031			X	
FU-032	X			
FU-033	X		X	
FU-034	X			X
FU-035	X			X
FU-036				X

FU-037				X
FU-038				X
FU-039	X			
FU-040	X			
FU-041	X			
FU-042	X			X
FU-043	X			X
FU-044	X			X
FU-045	X			X
FU-046	X			
FU-047	X			
FU-048	X			
FU-049	X			
FU-050				X

Tabla 14: Matriz de trazabilidad entre requisitos y escenarios.

5.2 Análisis de resultados

Los resultados obtenidos tras la realización de los cuatro escenarios propuestos son los que se exponen en la [Tabla 15](#). En ellos se puede comprobar que la solución final verifica todos los requisitos puesto que los usuarios han conseguido completar las tareas correspondientes de cada escenario de prueba.

Prueba	Fecha	Nº de intentos	Resultado	Observaciones
ESC-001	01/04/2013	1	Satisfactorio	
ESC-002	05/04/2013	1	Satisfactorio	
ESC-003	08/04/2013	1	Satisfactorio	
ESC-004	12/04/2013	1	Satisfactorio	

Tabla 15: Análisis de resultados de pruebas.

6 Conclusions

The purpose of this project is providing help to operators and the principal aim is the definition of the user interface of an operational system for the *Smart Grid*. In this part of memory the conclusions of the project are exposed. First of all, here is exposed the contributions in the project and further work. The problems encountered during development, also are explained in this section. Finally of this section describes the personal opinions.

6.1 Contributions

The principal challenge is the uncommon dimensions of screen. This contribution is different the use of three or four monitors in the rooms of control. This contribution is very import because it is basis for the following contributions. Owing to special dimensions of screen is necessary to break with the rooms of control paradigm. The first consequence of the dimensions of screen is the division of the screen into four zones. The division is the result of the needs of the operator because the operator needs to execute some actions simultaneously. It's a contribution because the current systems don't allow the operator to simultaneously execute actions. The second contribution and consequence of the dimensions of screen is widgets size. This is a challenge because the design of widgets for this dimensions screen change so much. The third consequence of the dimensions of screen is the coordination among of the four zones. This idea is very important because of the operator needs to use different points of view of an element for the control of the same one. The importance of this contribution resides in that each of the zones offers a point of view of an element. The last contribution is the own system with his features.

6.2 Further work

In the future this project could be extended in some areas. The aim of this project is to help the operator to simultaneously perform actions like monitoring or controlling; however, these actions could be hindered in such operational situations in which a huge amount of alarms are triggered. Then, a way to extend this project would be the creation of a crisis desk.

In a crisis situation the composition layers must change. The operators focus on a point of the grid. It is in this situation where the feasibility of the map gets less significant. The geographical map is located in the background. Conversely, graphics data are become more and more important. Graphics should be about the number of alarms and the typology of them. Additionally, graphics could be developed depending on the available spatial area.

Another extension of the project would be to improve the control zone. In this area you can add the capability of opening multiple actions simultaneously.

6.3 Problems

In the development of this project I have found several problems. The main one would be the shortcoming of the technology. Windows Presentation Foundation is a very special technology

because WPF needs to apply the pattern of design Model-View-ViewModel. On internet there is not so much information about the MVVM pattern, so the integration into the project was a challenging task. Due to this non-intuitive implementation model, most of the time was spent reading and reviewing MVVM tutorials.

The information offered by the system is multidimensional. In this direction, the system manages different information's layers. So, this application must to work with coordinates views in order to process different dimensions of information. In consequence, there are two problems.

The first problem refers to data refresh, a very common issue in data processing. In this case, three types of data refresh have been supported: visual refresh in the map, refresh data in the table in the zone of monitoring, and the refresh zones coordinated. When the operator select an element, all the views must to be refresh for show the different dimensions of the element's information.

The second problem, the ArcGIS library has been used to maps of this project. The problem regarding the ArcGIS library had to do with the need of inserting elements into the map. In addition, much of these elements are dynamics, so the system needs to get the coordinates of these elements in runtime.

6.4 Personal opinions

One of the main challenges of the project was to select a suitable technology that allows meeting systems requirements. In my opinion the project has been developed in the right technology. I believe that is the right technology because it offers much versatility for creating components. To support the operator in their tasks need interaction components. In addition, Windows Presentation Foundation doesn't constraint the design user's interfaces of application. Another aspect to consider is the use of a curved monitor. In this case I think it is positive.

Although the technology and the monitor are correct on this project, I believe there are some problems in the design. For example, alphanumeric representation should be intensively used in order to the displayed in the monitoring area. Another example is the display of each element of the map. For each type of element could change the icon. And this way the project would achieve greater realism.

Show the information is a common problem for a lot of monitoring systems. The solution presents has focused on interaction with the information's elements. So, in my opinion is a good option because the user interface doesn't have complex data component.

At last, with the development of the project I learned new technologies as Java RCP and Windows Presentation Foundation. Besides, I learned new design pattern of architecture Model-View-ViewModel.

7 Bibliografía

- [1] D. D. Cebollero, *Comparativa usabilidad de monitores*, Madrid, 2012.
- [2] "Microsoft," [Online]. Available: msdn.microsoft.com. [Accessed 3 abril 2013].
- [3] "WPF Tutorial," [Online]. Available: wpftutorial.net . [Accessed 3 abril 2013].
- [4] "Proyecto Energios," [Online]. Available: <http://innovationenergy.org/energios/>. [Accessed 5 abril 2013].
- [5] R. S. y. M. M. Armando Cabrera, "SlideShare," [Online]. Available: <http://www.slideshare.net/rfsolano/procesos-de-ingenieria-del-software>. [Accessed 16 marzo 2013].
- [6] D. H. Rothenberg, *Alarm Management for Process Control*, New Jersey: Monumentum Press, 2009.
- [7] Volere, "Plantilla de Especificación de Requisitos," Atalantic Systems Guild, 2006. [Online]. Available: http://www.volere.co.uk/pdf%20files/template_es.pdf. [Accessed 15 abril 2013].
- [8] Evolus, "Pencil," Evolus, [Online]. Available: <http://pencil.evolus.vn/>. [Accessed 2013 abril 28].
- [9] E. Foundation, "Eclipse," [Online]. Available: <http://www.eclipse.org/>.
- [10] Microsoft, "Microsoft," [Online]. Available: <http://www.microsoft.com/visualstudio/esn/products/visual-studio-overview>.
- [11] esri, "arcgis," esri, [Online]. Available: <http://www.arcgis.com/home/>.
- [12] Oracle, "MySQL," [Online]. Available: <http://www.mysql.com/>.
- [13] Microsoft, "Office Microsoft," [Online]. Available: <http://office.microsoft.com/es-es/excel/>.
- [14] P. M. Fitts, "The information capacity of the human motor system in controlling the amplitude of movement.," *Journal of Experimental Psychology*, vol. 47, no. 6, pp. 381-391, 1954.
- [15] J. a. M. R. Nielsen, "Heuristic evaluation of user interfaces," in *Proceedings of the ACM CHI 90 Human Factors in Computing Systems Conference*, Seattle, Washington, USA, 1990.

Anexo I. Control de versiones

En este anexo se incluye una tabla que establece el control de versiones que se ha llevado a lo largo del desarrollo de esta memoria.

Fecha	Versión	Contenido
12/04/2013	1.0	Desarrollo de los capítulos de introducción, estado del arte.
26/04/2013	1.1	Finalización del capítulo de estado del arte. Desarrollo del capítulo de gestión del proyecto.
05/05/2013	1.2	Desarrollo de la parte de análisis del capítulo de la solución. Y especificación de requisitos.
25/05/2013	1.3	Desarrollo de la parte de diseño e implementación del capítulo de la solución.
05/06/2013	1.4	Finalización de los capítulos de diseño e implementación. Desarrollo de la evaluación de la solución.
12/06/2013	1.5	Desarrollo del capítulo de conclusiones y resumen del trabajo.
13/06/2013	1.6	Desarrollo de Anexos.
14/06/2013	2.0	Inserción de imágenes y formato del documento.

Anexo II. Seguimiento de trabajo final de grado

En este anexo se expone la forma de seguimiento del proyecto por parte del tutor. También se expone la planificación final seguida para el desarrollo de la solución software y de la memoria correspondiente a la solución.

Forma de seguimiento

La manera de seguimiento llevada a cabo en la realización del proyecto se divide en dos tipos de reuniones: reuniones de seguimiento de desarrollo y reuniones de seguimiento de la memoria.

Las reuniones de seguimiento de desarrollo han sido programadas a una vez por semana. En estas reuniones el tutor orientaba los objetivos del diseño para llevar a cabo en tiempos determinados. El tutor también exponía los defectos que podrían encontrar en el trabajo desarrollado hasta ese momento o las posibles mejoras según sus conocimientos.

Las reuniones de seguimiento de la memoria han sido programadas cada quince días. En estas reuniones la tutora daba pautas que podría seguir a la hora de concretar los objetivos de cada sección del presente documento.

Planificación final

A continuación se muestra un diagrama de Gantt. Este diagrama especifica los tiempos utilizados para el desarrollo de cada uno de los prototipos software. La primera tarea que se muestra en el diagrama se corresponde con el estudio de las tecnologías. Esta tarea representa el tiempo que se dedicó al estudio de cada una de las tecnologías para las que se realizó un prototipo software. En el caso de *Windows presentation Foundation*, este tiempo se invirtió en el estudio del modelo arquitectónico Model-View-ViewModel. La otra tecnología estudiada es Java *Rich Client Platform*. Las tareas siguientes son el desarrollo de cada uno de los prototipos definidos en el apartado de diseño de este documento.

Id.	Nombre de tarea	Comienzo	Fin	Duración	ene 2013		feb 2013			mar 2013				abr 2013				may 2013						
					20/1	27/1	3/2	10/2	17/2	24/2	3/3	10/3	17/3	24/3	31/3	7/4	14/4	21/4	28/4	5/5	12/5	19/5	26/5	
1	Estudio de tecnologías	21/01/2013	01/02/2013	10d																				
2	Prototipo Java RCP	04/02/2013	15/02/2013	10d																				
3	Primer prototipo WPF	18/02/2013	22/03/2013	25d																				
4	Segundo prototipo WPF	25/03/2013	19/04/2013	20d																				
5	Solución final WPF	22/04/2013	31/05/2013	30d																				

Anexo III. Especificación de requisitos

En este anexo se lleva a cabo la especificación de todos los requisitos que se han listado en el apartado *Definición de Requisitos*. Para la especificación se sigue la plantilla detallada en la Tabla 9 del documento.

La especificación de requisitos se va a dividir en dos bloques: *funcionales* y *no funcionales*.

Requisitos funcionales

Identificador	FU-001		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario deberá comprender cuatro zonas: comunicación, monitorización, control y actuaciones.		
Prueba de Verificación	El operador será capaz de observar a la vez las tareas de cada zona diferenciada.		

Identificador	FU-002		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario mostrará una monitorización global de la red eléctrica en la zona de monitorización.		
Prueba de Verificación	El operador será capaz de monitorizar globalmente la red eléctrica.		

Identificador	FU-003		
Tipo de Requisito	Usuario	Prioridad	Media
Descripción	La interfaz de usuario permitirá cambiar la vista del mapa espacial de monitorización entre: relieve y callejero.		
Prueba de Verificación	El operador será capaz de seleccionar la vista del mapa espacial que desee.		

Identificador	FU-004		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá ampliar cualquier zona de la visión espacial con la función de la lupa.		
Prueba de Verificación	El operador será capaz de activar la función de lupa.		

Identificador	FU-005		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá consultar la información específica de cada alarma en la vista tabular.		

Prueba de Verificación	El operador será capaz de seleccionar una alarma en la vista tabular.
-------------------------------	---

Identificador	FU-006		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario mostrará la información de una alarma en un <i>tooltip</i> cerca de la alarma seleccionada.		
Prueba de Verificación	El operador será capaz de seleccionar una alarma en la vista espacial y consultar los detalles de la misma.		

Identificador	FU-007		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario permitirá mostrar el mapa posicional de la red eléctrica.		
Prueba de Verificación	El operador será capaz de cambiar la visión espacial por el mapa posicional de la red eléctrica.		

Identificador	FU-008		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario mostrará una vista tabular de datos alfanuméricos.		
Prueba de Verificación	El operador será capaz de consultar los datos de cualquier elemento en la vista tabular.		

Identificador	FU-009		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario tendrá una coordinación entre la vista tabular y los elementos mostrados en la visualización espacial.		
Prueba de Verificación	El operador será capaz de seleccionar un elemento en la vista tabular e identificar el elemento seleccionado en la monitorización de la red eléctrica.		

Identificador	FU-010		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	En la visualización espacial se ha de mostrar siempre la infraestructura de red.		
Prueba de Verificación	El operador no será capaz de desactivar la capa que muestra la infraestructura de red.		

Identificador	FU-011		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario tendrá que mostrar la infraestructura de red con diferentes colores según la zona de la red eléctrica que se trate.		
Prueba de Verificación	El operador podrá identificar las zonas de la red eléctrica según la agrupación de colores.		

Identificador	FU-012		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	Deberá existir una coordinación entre las 4 zonas de la interfaz de usuario.		
Prueba de Verificación	El operador será capaz observar un mismo elemento desde los diferentes cuatro puntos de vista: mensajes relacionados, mapa, ortogonal, etc.		

Identificador	FU-013		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario mostrará en la zona de comunicación, la comunicación asíncrona entre los operadores.		
Prueba de Verificación	El operador será capaz de ver los mensajes recibidos en la zona de comunicación.		

Identificador	FU-014		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá enviar mensajes desde la zona de comunicación.		
Prueba de Verificación	El operador será capaz de enviar un mensaje a otro operador.		

Identificador	FU-015		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá responder mensajes desde la zona de comunicación.		
Prueba de Verificación	El operador será capaz de responder un mensaje recibido en la zona de comunicación de la interfaz.		

Identificador	FU-016		
Tipo de Requisito	Sistema	Prioridad	Media
Descripción	El usuario podrá clasificar los distintos mensajes en la zona de comunicación.		
Prueba de	El operador será capaz de consultar diferentes tipos de mensajes:		

Verificación	enviados, recibidos, etc.
---------------------	---------------------------

Identificador	FU-017		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá seleccionar un área en la visión espacial de la red eléctrica que englobe todas las incidencias de esa zona concreta.		
Prueba de Verificación	El operador será capaz de dibujar el área que desee de la visión espacial.		

Identificador	FU-018		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá seleccionar una línea en la visión espacial de la red eléctrica que filtre las incidencias de ese tramo.		
Prueba de Verificación	El operador será capaz de dibujar un área que seleccione la línea incluida en el área.		

Identificador	FU-019		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	Cada tipo de elemento a mostrar en la visión espacial se albergará en una capa.		
Prueba de Verificación	El operador será capaz de relacionar las capas existentes con cada tipo de elemento que se encuentre en la red eléctrica.		

Identificador	FU-020		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	La interfaz de usuario permitirá al usuario activar o desactivar todas las capas posibles excepto la correspondiente a la infraestructura de la red.		
Prueba de Verificación	El operador será capaz de seleccionar y activar la capa que desee.		

Identificador	FU-021		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario indicará las alarmas que surjan nuevas con un parpadeo del icono.		
Prueba de Verificación	El operador será capaz de identificar las alarmas nuevas en la monitorización de la red eléctrica.		

Identificador	FU-022		
Tipo de Requisito	Usuario	Prioridad	Alta

Descripción	La interfaz de usuario marcará las alarmas nuevas en la vista tabular marcando la fila de la alarma con el color del icono de la misma.		
Prueba de Verificación	El operador será capaz de identificar las alarmas nuevas en la vista tabular y, además, será capaz de relacionar cada una con su icono correspondiente en la visión espacial.		

Identificador	FU-023		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario marcará las alarmas nuevas en la visión espacial con el icono sombreado y redondeado.		
Prueba de Verificación	El operador será capaz de identificar las alarmas nuevas en la monitorización de la red eléctrica.		

Identificador	FU-024		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	El color con el que se marcan las nuevas alarmas en la visión espacial perderá intensidad a medida que vaya pasando el tiempo hasta quedarse blancas.		
Prueba de Verificación	El operador será capaz de identificar el tiempo transcurrido desde que surgió la alarma.		

Identificador	FU-025		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá acceder a un menú contextual en los elementos de la infraestructura de red.		
Prueba de Verificación	El operador será capaz de desplegar el menú contextual.		

Identificador	FU-026		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El menú contextual de cada elemento tendrá las opciones: “Notas”, “Detalle” y “Histórico”.		
Prueba de Verificación	El operador será capaz de diferenciar las opciones.		

Identificador	FU-027		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La opción “Histórico” mostrará un gráfico de las medidas eléctricas tomadas en un rango de 10 segundos.		

Prueba de Verificación	El operador será capaz de consultar los gráficos de cada medida eléctrica tomada.
-------------------------------	---

Identificador	FU-028		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La opción “Detalle” mostrará una tabla con cada una de las características del elemento seleccionado.		
Prueba de Verificación	El operador será capaz de consultar los detalles del elemento de infraestructura seleccionado.		

Identificador	FU-029		
Tipo de Requisito	Usuario	Prioridad	Media
Descripción	La opción “Notas” mostrará el bloc de notas del elemento seleccionado.		
Prueba de Verificación	El operador será capaz de visualizar las notas sobre el elemento seleccionado de la infraestructura de red.		

Identificador	FU-030		
Tipo de Requisito	Sistema	Prioridad	Media
Descripción	La zona de control deberá mostrarse vacía por defecto.		
Prueba de Verificación	El operador observará, al iniciar la aplicación, que la zona de control se encuentra vacía.		

Identificador	FU-031		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	Al seleccionar una línea en la visión espacial, en la zona de control se mostrará: el mapa unifilar de la línea, un árbol de los componentes de la misma y un listado de incidencias del tramo.		
Prueba de Verificación	El operador será capaz de identificar las ventanas relativas de: unifilar de la línea seleccionada en el área, el árbol de componentes y el listado de incidencias.		

Identificador	FU-032		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	Al seleccionar un área en la visión espacial, en la zona de control se mostrará: el área seleccionada recortada del mapa, un listado de incidencias y la visión ortogonal del área correspondiente.		
Prueba de Verificación	El operador será capaz de identificar las ventanas relativas de: recorte del área seleccionada, visión ortogonal del área y el listado de incidencias.		

Identificador	FU-033		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	En la visión espacial deberán estar las capas correspondientes a: brigadas, avisos de clientes, puntos críticos, alarmas e infraestructura de red.		
Prueba de Verificación	El operador será capaz de identificar cada una de las capas.		

Identificador	FU-034		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La zona de actuaciones contendrá el listado de actuaciones del operador.		
Prueba de Verificación	El operador será capaz de localizar las actuaciones creadas.		

Identificador	FU-035		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	El listado de actuaciones estará clasificado por los siguientes tipos de actuaciones: programadas, no programadas, de otro y terminadas.		
Prueba de Verificación	El operador será capaz de diferenciar las actuaciones entre: programadas, no programadas, de otro y terminadas.		

Identificador	FU-036		
Tipo de Requisito	Sistema	Prioridad	Media
Descripción	El usuario desplegará una actuación en la zona de control mediante <i>DragAndDrop</i> .		
Prueba de Verificación	El operador será capaz de desplegar una actuación.		

Identificador	FU-037		
Tipo de Requisito	Sistema	Prioridad	Media
Descripción	El usuario podrá recoger la actuación ejecutando un <i>DragAndDrop</i> hacia la zona de actuaciones.		
Prueba de Verificación	El operador será capaz de recoger la actuación desplegada.		

Identificador	FU-038		
Tipo de Requisito	Usuario	Prioridad	Media

Descripción	El usuario podrá quitar todas las ventanas desplegadas en la zona de control con la opción “Limpiar”.
Prueba de Verificación	El operador será capaz de limpiar la zona de control.

Identificador	FU-039		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	Al seleccionar un área o una línea, se podrá crear una actuación mediante la opción “Crear” en la zona de control.		
Prueba de Verificación	El operador será capaz de crear una actuación.		

Identificador	FU-040		
Tipo de Requisito	Sistema	Prioridad	Media
Descripción	Al crear una actuación será del tipo no programada.		
Prueba de Verificación	El operador será capaz de identificar la nueva actuación como no programada.		

Identificador	FU-041		
Tipo de Requisito	Usuario	Prioridad	Media
Descripción	Al crear una actuación se iniciará un <i>log</i> de registros sobre esta, de las actividades realizadas en la actuación.		
Prueba de Verificación	El operador será capaz de identificar el <i>log</i> de registros de la actuación creada.		

Identificador	FU-042		
Tipo de Requisito	Usuario	Prioridad	Baja
Descripción	Las actuaciones programadas serán identificadas con el color azul.		
Prueba de Verificación	El operador será capaz de identificar las actuaciones programadas con el color azul.		

Identificador	FU-043		
Tipo de Requisito	Usuario	Prioridad	Baja
Descripción	Las actuaciones no programadas serán identificadas con el color rojo.		
Prueba de Verificación	El operador será capaz de identificar las actuaciones programadas con el color rojo.		

Identificador	FU-044		
Tipo de Requisito	Usuario	Prioridad	Baja

Descripción	Las actuaciones de otro serán identificadas con el color amarillo.
Prueba de Verificación	El operador será capaz de identificar las actuaciones programadas con el color amarilla.

Identificador	FU-045		
Tipo de Requisito	Usuario	Prioridad	Baja
Descripción	Las actuaciones terminadas serán identificadas con el color verde.		
Prueba de Verificación	El operador será capaz de identificar las actuaciones programadas con el color verde.		

Identificador	FU-046		
Tipo de Requisito	Usuario	Prioridad	Media
Descripción	Al seleccionar un área, se podrán activar o desactivar las capas de elementos en el recorte del área que se encuentra en la zona de control.		
Prueba de Verificación	El operador será capaz de activar y desactivar capas en el recorte del área que se muestra en la zona de control.		

Identificador	FU-047		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario permitirá mostrar la secuencia de aparición de alarmas desde el momento actual hasta 4 minutos atrás.		
Prueba de Verificación	El operador será capaz de visualizar las alarmas que han ocurrido hasta 4 minutos atrás.		

Identificador	FU-048		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario permitirá recrear una simulación de las alarmas que puedan aparecer desde el momento actual hasta 4 minutos en el futuro.		
Prueba de Verificación	El operador será capaz de visualizar la simulación de las alarmas que se pueden dar desde el momento actual hasta 4 minutos a posteriori.		

Identificador	FU-049		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario deberá indicar al usuario cuando está visualizando una simulación.		
Prueba de Verificación	El operador será consciente de que está visualizando una simulación.		

Identificador	FU-050		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El usuario podrá editar el <i>log</i> de registros de una actuación.		
Prueba de Verificación	El operador será capaz de editar registros del <i>log</i> de una actuación.		

Requisitos no funcionales

Identificador	NF-001		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El sistema se ejecutará en un monitor curvo de 42".		
Prueba de Verificación	El sistema no será capaz de mostrar la interfaz con las medidas correctas en un monitor con menos pulgadas que 42".		

Identificador	NF-002		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	El sistema será ejecutado sobre el sistema operativo Windows.		
Prueba de Verificación	El sistema no será capaz de ejecutar en ningún otro sistema operativo.		

Identificador	NF-003		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El sistema deberá actualizar la información cada 0,5 segundos.		
Prueba de Verificación	El sistema será capaz de recoger las alarmas nuevas que ocurran.		

Identificador	NF-004		
Tipo de Requisito	Usuario	Prioridad	Alta
Descripción	La interfaz de usuario seguirá el estándar Abnormal Situation Management (ASM) [6] para la clasificación de alarmas.		
Prueba de Verificación	El operador será capaz de reconocer los colores y formas establecidos por el estándar.		

Identificador	NF-005		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El sistema será una aplicación de escritorio.		
Prueba de Verificación	El sistema sólo podrá ejecutar como aplicación de escritorio.		

Identificador	NF-006		
Tipo de Requisito	Sistema	Prioridad	Alta
Descripción	El entorno de ejecución deberá tener instalado Microsoft Office 2010.		
Prueba de Verificación	El sistema será capaz de conectar con los archivos de Microsoft Office.		